



CÓD: OP-0930T-23
7908403543953

CRN-8

CONSELHO REGIONAL DE NUTRICIONISTAS 8ª REGIÃO

Assistente de Tecnologia da Informação Júnior

EDITAL Nº 1, DE 18 DE OUTUBRO DE 2023

Língua Portuguesa

1. Compreensão e interpretação de textos de gêneros variados.	7
2. Reconhecimento de tipos e gêneros textuais.	7
3. Domínio da ortografia oficial.	8
4. Domínio dos mecanismos de coesão textual. Emprego de elementos de referência, substituição e repetição, de conectores e de outros elementos de sequenciação textual.	9
5. Emprego de tempos e modos verbais. Emprego das classes de palavras.	9
6. Domínio da estrutura morfossintática do período.	16
7. Relações de coordenação entre orações e entre termos da oração. Relações de subordinação entre orações e entre termos da oração.	19
8. Emprego dos sinais de pontuação.	24
9. Concordância verbal e nominal.	27
10. Regência verbal e nominal.	29
11. Emprego do sinal indicativo de crase.	30
12. Colocação dos pronomes átonos.	30
13. Reescrita de frases e parágrafos do texto.	31
14. Significação das palavras.	37
15. Substituição de palavras ou de trechos de texto.	37
16. Reorganização da estrutura de orações e de períodos do texto.	38
17. Reescrita de textos de diferentes gêneros e níveis de formalidade.	38
18. Figuras de linguagem.	38
19. Correspondência oficial (conforme Manual de Redação da Presidência da República). Aspectos gerais da redação oficial. Finalidade dos expedientes oficiais.	41

Raciocínio Lógico e Matemática

1. Operações, propriedades e aplicações (soma, subtração, multiplicação, divisão, potenciação e radiciação).	57
2. Princípios de contagem e probabilidade.	63
3. Arranjos e permutações. Combinações.	65
4. Conjuntos numéricos (números naturais, inteiros, racionais e reais) e operações com conjuntos.	68
5. Razões e proporções (grandezas diretamente proporcionais, grandezas inversamente proporcionais, porcentagem, regras de três simples e compostas).	70
6. Equações e inequações.	74
7. Sistemas de medidas. Volumes.	78
8. Compreensão de estruturas lógicas.	80
9. Lógica de argumentação (analogias, inferências, deduções e conclusões).	80
10. Diagramas lógicos.	84

Noções de Informática

1. Conceitos básicos e modos de utilização de tecnologias, ferramentas, aplicativos e procedimentos de informática: tipos de computador, conceitos de hardware e de software, instalação de periféricos.	89
2. Edição de textos, planilhas e apresentações (ambiente Microsoft Office, versões 2016, 2021 e 365).	90
3. Noções de sistema operacional (ambiente Windows, versões 10 e 11).	95
4. Redes de computadores: conceitos básicos, ferramentas, aplicativos e procedimentos de Internet e intranet. Programas de navegação: Mozilla Firefox, Google Chrome e Microsoft Edge. Programa de correio eletrônico: MS Outlook. Sítios de busca e pesquisa na Internet.	97
5. Conceitos de organização e de gerenciamento de informações, arquivos, pastas e programas.	105
6. Segurança da informação: procedimentos de segurança. Noções de vírus, worms e pragas virtuais. Conceitos de Spam e phishing. Aplicativos para segurança (antivírus, firewall, antispyware etc.). Procedimentos de backup.	107

Atualidades

1. Tópicos atuais e relevantes de diversas áreas, tais como segurança, transportes, política, economia, sociedade, educação, saúde, cultura, tecnologia, energia, relações internacionais, desenvolvimento sustentável e ecologia.	111
---	-----

Conhecimentos Básicos em Hardware e Software

1. Conceitos, fundamentos, montagem e manutenção de hardware e software; sistemas operacionais.	113
2. MS-Windows 10/11 e Linux.	113

Redes de Computadores e Segurança

1. Conceitos, fundamentos, arquitetura, meios de transmissão;	117
2. redes LAN e WAN;	118
3. políticas de backup/restore; anti-vírus	118

Análise e Desenvolvimento de Sistemas

1. Lógica de programação: algoritmos, fluxogramas, programação orientada a objetos, depuração.	119
2. Organização da informação: estruturas de dados, arquivos e registros, bancos de dados.	120

Linguagens de banco de dados

1. SQL. Banco de dados: fundamentos de gerência de banco de dados relacional, conceitos, componentes, funcionamento, segurança e integridade, código armazenado (stored procedures e triggers) (My-SQL, ORACLE).	125
2. Internet/Intranet: Correio eletrônico: IMAP, SMTP, controle de SPAM.	129
3. Serviços de diretórios e LDAP	131

Servidores web:

1. Apache.	133
2. Tecnologias web: AJAX, Webservices (XML, RSS), CSS, DHTML	133

Linguagens de programação para web

1. php, javascript, HTML	137
--------------------------------	-----

Conteúdo Digital:

Legislação e Ética na Administração Pública

1. Ética e função pública	3
2. Ética no Setor Público	5
3. Lei nº 8.429/1992 e suas alterações	6
4. Lei nº 9.784/1999 e suas alterações (processo administrativo).....	15
5. Acesso à informação: Lei nº 12.527/2011	20
6. Decreto nº 7.724/2012	27
7. Decreto nº 9.830/2019	38

Legislação específica

1. Lei Federal nº 6.583/1978.....	43
2. Lei Federal nº 6.839/1980.....	45
3. Lei Federal nº 8.234/1991.....	45
4. Lei Federal nº 13.709/2018.....	46
5. Lei Federal nº 14.133/2021.....	59
6. Decreto nº 84.444/1980	100
7. Resolução CFN: nº 733/2022	105
8. Resolução CFN: nº 705/2021	106
9. Resolução CFN: nº 600/2018	134
10. Resolução CFN: nº 599/2018	164
11. Resolução CFN: nº 466/2010	172
12. Resolução CFN: nº 465/2010	175
13. Resolução CFN: nº 356/2004	178

Conteúdo Digital

- Para estudar o Conteúdo Digital acesse sua “Área do Cliente” em nosso site, ou siga os passos indicados na página 2 para acessar seu bônus.

<https://www.apostilasopcao.com.br/customer/account/login/>

LÍNGUA PORTUGUESA

COMPREENSÃO E INTERPRETAÇÃO DE TEXTOS DE GÊNEROS VARIADOS.

Compreender e interpretar textos é essencial para que o objetivo de comunicação seja alcançado satisfatoriamente. Com isso, é importante saber diferenciar os dois conceitos. Vale lembrar que o texto pode ser verbal ou não-verbal, desde que tenha um sentido completo.

A **compreensão** se relaciona ao entendimento de um texto e de sua proposta comunicativa, decodificando a mensagem explícita. Só depois de compreender o texto que é possível fazer a sua interpretação.

A **interpretação** são as conclusões que chegamos a partir do conteúdo do texto, isto é, ela se encontra para além daquilo que está escrito ou mostrado. Assim, podemos dizer que a interpretação é subjetiva, contando com o conhecimento prévio e do repertório do leitor.

Dessa maneira, para compreender e interpretar bem um texto, é necessário fazer a decodificação de códigos linguísticos e/ou visuais, isto é, identificar figuras de linguagem, reconhecer o sentido de conjunções e preposições, por exemplo, bem como identificar expressões, gestos e cores quando se trata de imagens.

Dicas práticas

1. Faça um resumo (pode ser uma palavra, uma frase, um conceito) sobre o assunto e os argumentos apresentados em cada parágrafo, tentando traçar a linha de raciocínio do texto. Se possível, adicione também pensamentos e inferências próprias às anotações.

2. Tenha sempre um dicionário ou uma ferramenta de busca por perto, para poder procurar o significado de palavras desconhecidas.

3. Fique atento aos detalhes oferecidos pelo texto: dados, fonte de referências e datas.

4. Sublinhe as informações importantes, separando fatos de opiniões.

5. Perceba o enunciado das questões. De um modo geral, questões que esperam **compreensão do texto** aparecem com as seguintes expressões: *o autor afirma/sugere que...; segundo o texto...; de acordo com o autor...* Já as questões que esperam **interpretação do texto** aparecem com as seguintes expressões: *conclui-se do texto que...; o texto permite deduzir que...; qual é a intenção do autor quando afirma que...*

RECONHECIMENTO DE TIPOS E GÊNEROS TEXTUAIS.

A partir da estrutura linguística, da função social e da finalidade de um texto, é possível identificar a qual tipo e gênero ele pertence. Antes, é preciso entender a diferença entre essas duas classificações.

Tipos textuais

A tipologia textual se classifica a partir da estrutura e da finalidade do texto, ou seja, está relacionada ao modo como o texto se apresenta. A partir de sua função, é possível estabelecer um padrão específico para se fazer a enunciação.

Veja, no quadro abaixo, os principais tipos e suas características:

TEXTO NARRATIVO	Apresenta um enredo, com ações e relações entre personagens, que ocorre em determinados espaço e tempo. É contado por um narrador, e se estrutura da seguinte maneira: apresentação > desenvolvimento > clímax > desfecho
TEXTO DISSERTATIVO-ARGUMENTATIVO	Tem o objetivo de defender determinado ponto de vista, persuadindo o leitor a partir do uso de argumentos sólidos. Sua estrutura comum é: introdução > desenvolvimento > conclusão.
TEXTO EXPOSITIVO	Procura expor ideias, sem a necessidade de defender algum ponto de vista. Para isso, usa-se comparações, informações, definições, conceitualizações etc. A estrutura segue a do texto dissertativo-argumentativo.
TEXTO DESCRITIVO	Expõe acontecimentos, lugares, pessoas, de modo que sua finalidade é descrever, ou seja, caracterizar algo ou alguém. Com isso, é um texto rico em adjetivos e em verbos de ligação.
TEXTO INJUNTIVO	Oferece instruções, com o objetivo de orientar o leitor. Sua maior característica são os verbos no modo imperativo.

Gêneros textuais

A classificação dos gêneros textuais se dá a partir do reconhecimento de certos padrões estruturais que se constituem a partir da função social do texto. No entanto, sua estrutura e seu estilo não são tão limitados e definidos como ocorre na tipologia textual, podendo se apresentar com uma grande diversidade. Além disso, o padrão também pode sofrer modificações ao longo do tempo, assim como a própria língua e a comunicação, no geral.

Alguns exemplos de gêneros textuais:

- Artigo
- Bilhete
- Bula
- Carta

- Conto
- Crônica
- E-mail
- Lista
- Manual
- Notícia
- Poema
- Propaganda
- Receita culinária
- Resenha
- Seminário

Vale lembrar que é comum enquadrar os gêneros textuais em determinados tipos textuais. No entanto, nada impede que um texto literário seja feito com a estruturação de uma receita culinária, por exemplo. Então, fique atento quanto às características, à finalidade e à função social de cada texto analisado.

DOMÍNIO DA ORTOGRAFIA OFICIAL.

A ortografia oficial diz respeito às regras gramaticais referentes à escrita correta das palavras. Para melhor entendê-las, é preciso analisar caso a caso. Lembre-se de que a melhor maneira de memorizar a ortografia correta de uma língua é por meio da leitura, que também faz aumentar o vocabulário do leitor.

Neste capítulo serão abordadas regras para dúvidas frequentes entre os falantes do português. No entanto, é importante ressaltar que existem inúmeras exceções para essas regras, portanto, fique atento!

Alfabeto

O primeiro passo para compreender a ortografia oficial é conhecer o alfabeto (os sinais gráficos e seus sons). No português, o alfabeto se constitui 26 letras, divididas entre **vogais** (a, e, i, o, u) e **consoantes** (restante das letras).

Com o Novo Acordo Ortográfico, as consoantes **K**, **W** e **Y** foram reintroduzidas ao alfabeto oficial da língua portuguesa, de modo que elas são usadas apenas em duas ocorrências: **transcrição de nomes próprios** e **abreviaturas e símbolos de uso internacional**.

Uso do “X”

Algumas dicas são relevantes para saber o momento de usar o X no lugar do CH:

- Depois das sílabas iniciais “me” e “en” (ex: mexerica; enxergar)
- Depois de ditongos (ex: caixa)
- Palavras de origem indígena ou africana (ex: abacaxi; orixá)

Uso do “S” ou “Z”

Algumas regras do uso do “S” com som de “Z” podem ser observadas:

- Depois de ditongos (ex: coisa)
- Em palavras derivadas cuja palavra primitiva já se usa o “S” (ex: casa > casinha)
- Nos sufixos “ês” e “esa”, ao indicarem nacionalidade, título ou origem. (ex: portuguesa)
- Nos sufixos formadores de adjetivos “ense”, “oso” e “osa” (ex: populoso)

Uso do “S”, “SS”, “Ç”

- “S” costuma aparecer entre uma vogal e uma consoante (ex: diversão)
- “SS” costuma aparecer entre duas vogais (ex: processo)
- “Ç” costuma aparecer em palavras estrangeiras que passaram pelo processo de aportuguesamento (ex: muçarela)

Os diferentes porquês

POR QUE	Usado para fazer perguntas. Pode ser substituído por “por qual motivo”
PORQUE	Usado em respostas e explicações. Pode ser substituído por “pois”
POR QUÊ	O “que” é acentuado quando aparece como a última palavra da frase, antes da pontuação final (interrogação, exclamação, ponto final)
PORQUÊ	É um substantivo, portanto costuma vir acompanhado de um artigo, numeral, adjetivo ou pronome

Parônimos e homônimos

As palavras **parônimas** são aquelas que possuem grafia e pronúncia semelhantes, porém com significados distintos.

Ex: *cumprimento* (saudação) X *comprimento* (extensão); *tráfego* (trânsito) X *tráfico* (comércio ilegal).

Já as palavras **homônimas** são aquelas que possuem a mesma grafia e pronúncia, porém têm significados diferentes. **Ex:** *rio* (verbo “rir”) X *rio* (curso d’água); *manga* (blusa) X *manga* (fruta).

DOMÍNIO DOS MECANISMOS DE COESÃO TEXTUAL. EMPREGO DE ELEMENTOS DE REFERENCIAÇÃO, SUBSTITUIÇÃO E REPETIÇÃO, DE CONECTORES E DE OUTROS ELEMENTOS DE SEQUENCIAÇÃO TEXTUAL.

A coerência e a coesão são essenciais na escrita e na interpretação de textos. Ambos se referem à relação adequada entre os componentes do texto, de modo que são independentes entre si. Isso quer dizer que um texto pode estar coeso, porém incoerente, e vice-versa.

Enquanto a coesão tem foco nas questões gramaticais, ou seja, ligação entre palavras, frases e parágrafos, a coerência diz respeito ao conteúdo, isto é, uma sequência lógica entre as ideias.

Coesão

A coesão textual ocorre, normalmente, por meio do uso de **conectivos** (preposições, conjunções, advérbios). Ela pode ser obtida a partir da **anáfora** (retoma um componente) e da **catáfora** (antecipa um componente).

Confira, então, as principais regras que garantem a coesão textual:

REGRA	CARACTERÍSTICAS	EXEMPLOS
REFERÊNCIA	Pessoal (uso de pronomes pessoais ou possessivos) – anafórica Demonstrativa (uso de pronomes demonstrativos e advérbios) – catafórica Comparativa (uso de comparações por semelhanças)	João e Maria são crianças. <i>Eles</i> são irmãos. Fiz todas as tarefas, exceto <i>esta</i> : colonização africana. Mais um ano <i>igual aos</i> outros...
SUBSTITUIÇÃO	Substituição de um termo por outro, para evitar repetição	Maria está triste. <i>A menina</i> está cansada de ficar em casa.
ELIPSE	Omissão de um termo	No quarto, apenas quatro ou cinco convidados. (omissão do verbo “haver”)
CONJUNÇÃO	Conexão entre duas orações, estabelecendo relação entre elas	Eu queria ir ao cinema, <i>mas</i> estamos de quarentena.
COESÃO LEXICAL	Utilização de sinônimos, hiperônimos, nomes genéricos ou palavras que possuem sentido aproximado e pertencente a um mesmo grupo lexical.	A minha <i>casa</i> é clara. Os <i>quartos</i> , a <i>sala</i> e a <i>cozinha</i> têm janelas grandes.

Coerência

Nesse caso, é importante conferir se a mensagem e a conexão de ideias fazem sentido, e seguem uma linha clara de raciocínio.

Existem alguns conceitos básicos que ajudam a garantir a coerência. Veja quais são os principais princípios para um texto coerente:

- **Princípio da não contradição:** não deve haver ideias contraditórias em diferentes partes do texto.
- **Princípio da não tautologia:** a ideia não deve estar redundante, ainda que seja expressa com palavras diferentes.
- **Princípio da relevância:** as ideias devem se relacionar entre si, não sendo fragmentadas nem sem propósito para a argumentação.
- **Princípio da continuidade temática:** é preciso que o assunto tenha um seguimento em relação ao assunto tratado.
- **Princípio da progressão semântica:** inserir informações novas, que sejam ordenadas de maneira adequada em relação à progressão de ideias.

Para atender a todos os princípios, alguns fatores são recomendáveis para garantir a coerência textual, como amplo **conhecimento de mundo**, isto é, a bagagem de informações que adquirimos ao longo da vida; **inferências** acerca do conhecimento de mundo do leitor; e **informatividade**, ou seja, conhecimentos ricos, interessantes e pouco previsíveis.

EMPREGO DE TEMPOS E MODOS VERBAIS. EMPREGO DAS CLASSES DE PALAVRAS.

Para entender sobre a estrutura das funções sintáticas, é preciso conhecer as classes de palavras, também conhecidas por classes morfológicas. A gramática tradicional pressupõe 10 classes gramaticais de palavras, sendo elas: adjetivo, advérbio, artigo, conjunção, interjeição, numeral, pronome, preposição, substantivo e verbo.

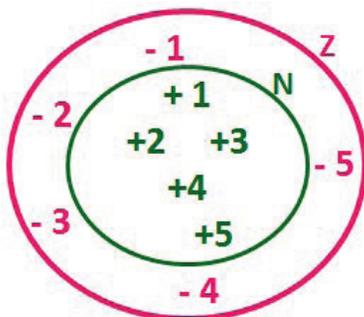
Veja, a seguir, as características principais de cada uma delas.

RACIOCÍNIO LÓGICO E MATEMÁTICA

OPERAÇÕES, PROPRIEDADES E APLICAÇÕES (SOMA, SUBTRAÇÃO, MULTIPLICAÇÃO, DIVISÃO, POTENCIAÇÃO E RADICIAÇÃO).

Conjunto dos números inteiros - z

O conjunto dos números inteiros é a reunião do conjunto dos números naturais $N = \{0, 1, 2, 3, 4, \dots, n, \dots\}$, $(N \subset Z)$; o conjunto dos opostos dos números naturais e o zero. Representamos pela letra Z.



$N \subset Z$ (N está contido em Z)

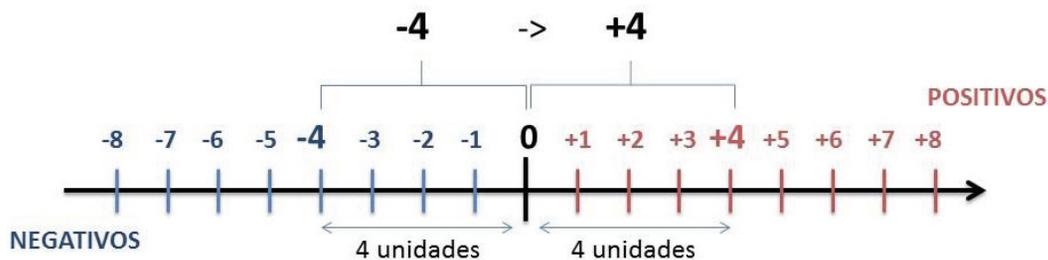
Subconjuntos:

SÍMBOLO	REPRESENTAÇÃO	DESCRIÇÃO
*	Z^*	Conjunto dos números inteiros não nulos
+	Z_+	Conjunto dos números inteiros não negativos
* e +	Z^*_+	Conjunto dos números inteiros positivos
-	Z_-	Conjunto dos números inteiros não positivos
* e -	Z^*_-	Conjunto dos números inteiros negativos

Observamos nos números inteiros algumas características:

- **Módulo:** distância ou afastamento desse número até o zero, na reta numérica inteira. Representa-se o módulo por $| \cdot |$. O módulo de qualquer número inteiro, diferente de zero, é sempre positivo.

- **Números Opostos:** dois números são opostos quando sua soma é zero. Isto significa que eles estão a mesma distância da origem (zero).



Somando-se temos: $(+4) + (-4) = (-4) + (+4) = 0$

Operações

• **Soma ou Adição:** Associamos aos números inteiros positivos a ideia de ganhar e aos números inteiros negativos a ideia de perder.

ATENÇÃO: O sinal (+) antes do número positivo pode ser dispensado, mas o sinal (-) antes do número negativo nunca pode ser dispensado.

• **Subtração:** empregamos quando precisamos tirar uma quantidade de outra quantidade; temos duas quantidades e queremos saber quanto uma delas tem a mais que a outra; temos duas quantidades e queremos saber quanto falta a uma delas para atingir a outra. A subtração é a operação inversa da adição. O sinal sempre será do maior número.

ATENÇÃO: todos parênteses, colchetes, chaves, números, ..., entre outros, precedidos de sinal negativo, tem o seu sinal invertido, ou seja, é dado o seu oposto.

Exemplo:

(FUNDAÇÃO CASA – AGENTE EDUCACIONAL – VUNESP) Para zelar pelos jovens internados e orientá-los a respeito do uso adequado dos materiais em geral e dos recursos utilizados em atividades educativas, bem como da preservação predial, realizou-se uma dinâmica elencando “atitudes positivas” e “atitudes negativas”, no entendimento dos elementos do grupo. Solicitou-se que cada um classificasse suas atitudes como positiva ou negativa, atribuindo (+4) pontos a cada atitude positiva e (-1) a cada atitude negativa. Se um jovem classificou como positiva apenas 20 das 50 atitudes anotadas, o total de pontos atribuídos foi

- (A) 50.
- (B) 45.
- (C) 42.
- (D) 36.
- (E) 32.

Resolução:

50-20=30 atitudes negativas
20.4=80
30.(-1)=-30
80-30=50

Resposta: A

• **Multiplicação:** é uma adição de números/ fatores repetidos. Na multiplicação o produto dos números *a* e *b*, pode ser indicado por ***a x b***, ***a . b*** ou ainda ***ab*** sem nenhum sinal entre as letras.

• **Divisão:** a divisão exata de um número inteiro por outro número inteiro, diferente de zero, dividimos o módulo do dividendo pelo módulo do divisor.

ATENÇÃO:

1) No conjunto *Z*, a divisão não é comutativa, não é associativa e não tem a propriedade da existência do elemento neutro.

2) Não existe divisão por zero.

3) Zero dividido por qualquer número inteiro, diferente de zero, é zero, pois o produto de qualquer número inteiro por zero é igual a zero.

Na multiplicação e divisão de números inteiros é muito importante a **REGRA DE SINAIS:**

Sinais iguais (+) (+); (-) (-) = resultado sempre positivo.

Sinais diferentes (+) (-); (-) (+) = resultado sempre negativo.

Exemplo:

(PREF.DE NITERÓI) Um estudante empilhou seus livros, obtendo uma única pilha 52cm de altura. Sabendo que 8 desses livros possui uma espessura de 2cm, e que os livros restantes possuem espessura de 3cm, o número de livros na pilha é:

- (A) 10
- (B) 15
- (C) 18
- (D) 20
- (E) 22

Resolução:

São 8 livros de 2 cm: $8 \cdot 2 = 16$ cm

Como eu tenho 52 cm ao todo e os demais livros tem 3 cm, temos:

$52 - 16 = 36$ cm de altura de livros de 3 cm

$36 : 3 = 12$ livros de 3 cm

O total de livros da pilha: $8 + 12 = 20$ livros ao todo.

Resposta: D

• **Potenciação:** A potência a^n do número inteiro *a*, é definida como um produto de *n* fatores iguais. O número *a* é denominado a **base** e o número *n* é o **expoente**. $a^n = a \times a \times a \times a \times \dots \times a$, *a* é multiplicado por *a* *n* vezes. Tenha em mente que:

– Toda potência de **base positiva** é um número **inteiro positivo**.

– Toda potência de **base negativa** e **expoente par** é um número **inteiro positivo**.

– Toda potência de **base negativa** e **expoente ímpar** é um número **inteiro negativo**.

Propriedades da Potenciação

1) Produtos de Potências com bases iguais: Conserva-se a base e somam-se os expoentes. $(-a)^3 \cdot (-a)^6 = (-a)^{3+6} = (-a)^9$

2) Quocientes de Potências com bases iguais: Conserva-se a base e subtraem-se os expoentes. $(-a)^8 : (-a)^6 = (-a)^{8-6} = (-a)^2$

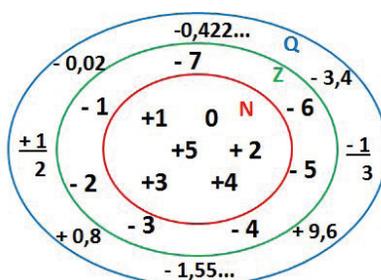
3) Potência de Potência: Conserva-se a base e multiplicam-se os expoentes. $[(-a)^5]^2 = (-a)^{5 \cdot 2} = (-a)^{10}$

4) Potência de expoente 1: É sempre igual à base. $(-a)^1 = -a$ e $(+a)^1 = +a$

5) Potência de expoente zero e base diferente de zero: É igual a 1. $(+a)^0 = 1$ e $(-b)^0 = 1$

Conjunto dos números racionais – Q

Um número racional é o que pode ser escrito na forma $\frac{m}{n}$, onde *m* e *n* são números inteiros, sendo que *n* deve ser diferente de zero. Frequentemente usamos *m/n* para significar a divisão de *m* por *n*.



N C Z C Q (N está contido em Z que está contido em Q)

Subconjuntos:

SÍMBOLO	REPRESENTAÇÃO	DESCRIÇÃO
*	Q^*	Conjunto dos números racionais não nulos
+	Q_+	Conjunto dos números racionais não negativos
* e +	Q^*_+	Conjunto dos números racionais positivos
-	Q_-	Conjunto dos números racionais não positivos
* e -	Q^*_-	Conjunto dos números racionais negativos

Representação decimal

Podemos representar um número racional, escrito na forma de fração, em número decimal. Para isso temos duas maneiras possíveis:

1ª) O numeral decimal obtido possui, após a vírgula, um número finito de algarismos. Decimais Exatos:

$$\frac{2}{5} = 0,4$$

2ª) O numeral decimal obtido possui, após a vírgula, infinitos algarismos (nem todos nulos), repetindo-se periodicamente Decimais Periódicos ou Dízimas Periódicas:

$$\frac{1}{3} = 0,333...$$

Representação Fracionária

É a operação inversa da anterior. Aqui temos duas maneiras possíveis:

1) Transformando o número decimal em uma fração numerador é o número decimal sem a vírgula e o denominador é composto pelo numeral 1, seguido de tantos zeros quantas forem as casas decimais do número decimal dado. Ex.:

$$0,035 = 35/1000$$

2) Através da fração geratriz. Aí temos o caso das dízimas periódicas que podem ser simples ou compostas.

– *Simplex*: o seu período é composto por um mesmo número ou conjunto de números que se repete infinitamente. Exemplos:

<p>* 0,444... Período: 4 (1 algarismo)</p> <p>$0,444... = \frac{4}{9}$</p>	<p>* 0,313131... Período: 31 (2 algarismos)</p> <p>$0,313131... = \frac{31}{99}$</p>	<p>* 0,278278278... Período: 278 (3 algarismos)</p> <p>$0,278278278... = \frac{278}{999}$</p>
---	---	--

Procedimento: para transformarmos uma dízima periódica simples em fração basta utilizarmos o dígito 9 no denominador para cada quantos dígitos tiver o período da dízima.

NOÇÕES DE INFORMÁTICA

CONCEITOS BÁSICOS E MODOS DE UTILIZAÇÃO DE TECNOLOGIAS, FERRAMENTAS, APLICATIVOS E PROCEDIMENTOS DE INFORMÁTICA: TIPOS DE COMPUTADOR, CONCEITOS DE HARDWARE E DE SOFTWARE, INSTALAÇÃO DE PERIFÉRICOS.

Hardware

Hardware refere-se a parte física do computador, isto é, são os dispositivos eletrônicos que necessitamos para usarmos o computador. Exemplos de hardware são: CPU, teclado, mouse, disco rígido, monitor, scanner, etc.

Software

Software, na verdade, são os programas usados para fazer tarefas e para fazer o hardware funcionar. As instruções de software são programadas em uma linguagem de computador, traduzidas em linguagem de máquina e executadas por computador.

O software pode ser categorizado em dois tipos:

- Software de sistema operacional
- Software de aplicativos em geral

• Software de sistema operacional

O software de sistema é o responsável pelo funcionamento do computador, é a plataforma de execução do usuário. Exemplos de software do sistema incluem sistemas operacionais como Windows, Linux, Unix, Solaris etc.

• Software de aplicação

O software de aplicação é aquele utilizado pelos usuários para execução de tarefas específicas. Exemplos de software de aplicativos incluem Microsoft Word, Excel, PowerPoint, Access, etc.

Para não esquecer:

HARDWARE	É a parte física do computador
SOFTWARE	São os programas no computador (de funcionamento e tarefas)

Periféricos

Periféricos são os dispositivos externos para serem utilizados no computador, ou mesmo para aprimorá-lo nas suas funcionalidades. Os dispositivos podem ser essenciais, como o teclado, ou aqueles que podem melhorar a experiência do usuário e até mesmo melhorar o desempenho do computador, tais como design, qualidade de som, alto falantes, etc.

Tipos:

PERIFÉRICOS DE ENTRADA	Utilizados para a entrada de dados;
PERIFÉRICOS DE SAÍDA	Utilizados para saída/visualização de dados

• Periféricos de entrada mais comuns.

- O teclado é o dispositivo de entrada mais popular e é um item essencial. Hoje em dia temos vários tipos de teclados ergonômicos para ajudar na digitação e evitar problemas de saúde muscular;
- Na mesma categoria temos o scanner, que digitaliza dados para uso no computador;
- O mouse também é um dispositivo importante, pois com ele podemos apontar para um item desejado, facilitando o uso do computador.

• Periféricos de saída populares mais comuns

- Monitores, que mostra dados e informações ao usuário;
- Impressoras, que permite a impressão de dados para material físico;
- Alto-falantes, que permitem a saída de áudio do computador;
- Fones de ouvido.

Sistema Operacional

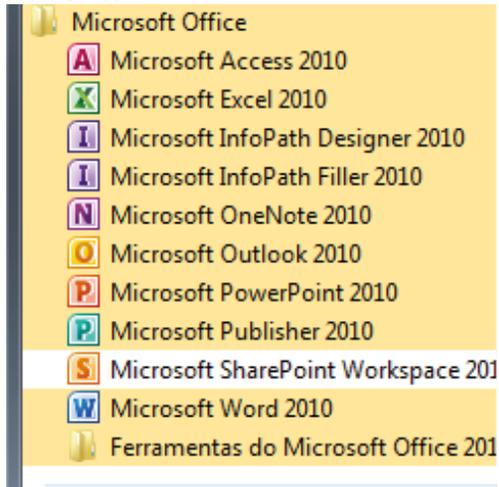
O software de sistema operacional é o responsável pelo funcionamento do computador. É a plataforma de execução do usuário. Exemplos de software do sistema incluem sistemas operacionais como Windows, Linux, Unix, Solaris etc.

• Aplicativos e Ferramentas

São softwares utilizados pelos usuários para execução de tarefas específicas. Exemplos: Microsoft Word, Excel, PowerPoint, Access, além de ferramentas construídas para fins específicos.

**EDIÇÃO DE TEXTOS, PLANILHAS E APRESENTAÇÕES
(AMBIENTE MICROSOFT OFFICE, VERSÕES 2016, 2021
E 365).**

Microsoft Office



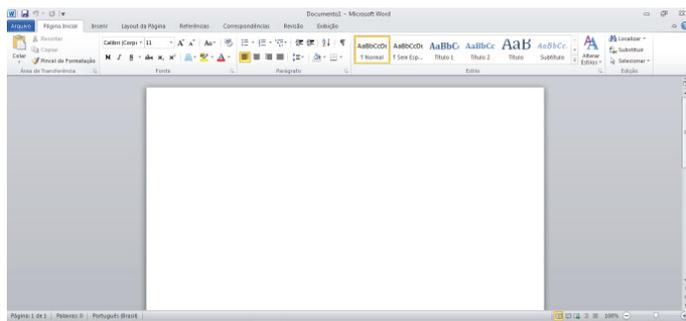
O Microsoft Office é um conjunto de aplicativos essenciais para uso pessoal e comercial, ele conta com diversas ferramentas, mas em geral são utilizadas e cobradas em provas o Editor de Textos – Word, o Editor de Planilhas – Excel, e o Editor de Apresentações – PowerPoint. A seguir verificamos sua utilização mais comum:

Word

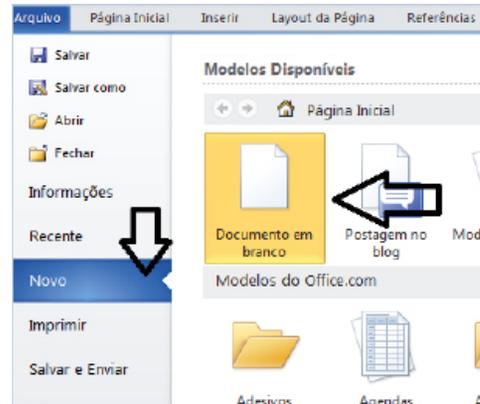
O Word é um editor de textos amplamente utilizado. Com ele podemos redigir cartas, comunicações, livros, apostilas, etc. Vamos então apresentar suas principais funcionalidades.

• Área de trabalho do Word

Nesta área podemos digitar nosso texto e formata-lo de acordo com a necessidade.



• Iniciando um novo documento



A partir deste botão retornamos para a área de trabalho do Word, onde podemos digitar nossos textos e aplicar as formatações desejadas.

• Alinhamentos

Ao digitar um texto, frequentemente temos que alinhá-lo para atender às necessidades. Na tabela a seguir, verificamos os alinhamentos automáticos disponíveis na plataforma do Word.

GUIA PÁGINA INICIAL	ALINHAMENTO	TECLA DE ATALHO
	Justificar (arruma a direita e a esquerda de acordo com a margem)	Ctrl + J
	Alinhamento à direita	Ctrl + G
	Centralizar o texto	Ctrl + E
	Alinhamento à esquerda	Ctrl + Q

• Formatação de letras (Tipos e Tamanho)

Presente em *Fonte*, na área de ferramentas no topo da área de trabalho, é neste menu que podemos formatar os aspectos básicos de nosso texto. Bem como: tipo de fonte, tamanho (ou pontuação), se será maiúscula ou minúscula e outros itens nos recursos automáticos.



GUIA PÁGINA INICIAL	FUNÇÃO
	Tipo de letra
	Tamanho
	Aumenta / diminui tamanho

	Recursos automáticos de caixa-altas e baixas
	Limpa a formatação

• **Marcadores**

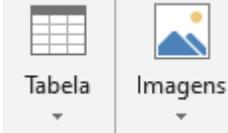
Muitas vezes queremos organizar um texto em tópicos da seguinte forma:

- **Item 1**
- **Item 2**
- **Item 2**

Podemos então utilizar na página inicial os botões para operar diferentes tipos de marcadores automáticos:



• Outros Recursos interessantes:

GUIA	ÍCONE	FUNÇÃO
Página inicial		- Mudar Forma - Mudar cor de Fundo - Mudar cor do texto
Inserir		- Inserir Tabelas - Inserir Imagens
Revisão		Verificação e correção ortográfica
Arquivo		Salvar

Excel

O Excel é um editor que permite a criação de tabelas para cálculos automáticos, análise de dados, gráficos, totais automáticos, dentre outras funcionalidades importantes, que fazem parte do dia a dia do uso pessoal e empresarial.

São exemplos de planilhas:

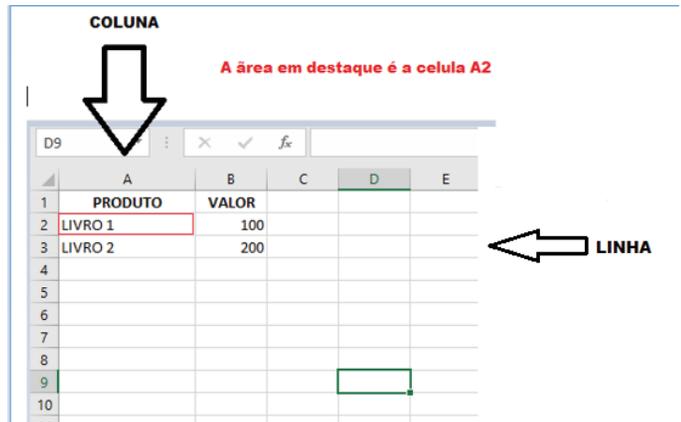
- Planilha de vendas;
- Planilha de custos.

Desta forma ao inserirmos dados, os valores são calculados automaticamente.

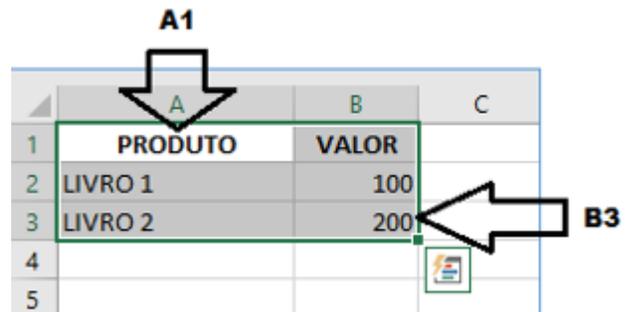
• **Mas como é uma planilha de cálculo?**

- Quando inseridos em alguma célula da planilha, os dados são calculados automaticamente mediante a aplicação de fórmulas específicas do aplicativo.

- A unidade central do Excel nada mais é que o cruzamento entre a linha e a coluna. No exemplo coluna A, linha 2 (A2)



- Podemos também ter o intervalo A1..B3



- Para inserirmos dados, basta posicionarmos o cursor na célula, selecionarmos e digitarmos. Assim se dá a iniciação básica de uma planilha.

• **Formatação células**

CONHECIMENTOS BÁSICOS EM HARDWARE E SOFTWARE:

CONCEITOS, FUNDAMENTOS, MONTAGEM E MANUTENÇÃO DE HARDWARE E SOFTWARE; SISTEMAS OPERACIONAIS.

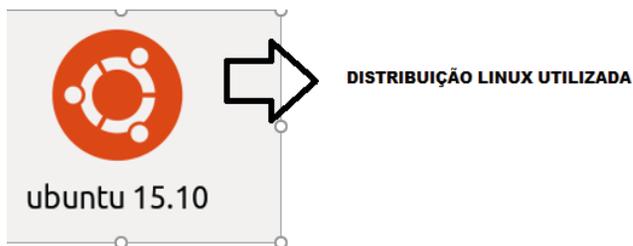
Prezado Candidato, o tema acima supracitado, já foi abordado em tópicos anteriores.

MS-WINDOWS 10/11 E LINUX

Prezado Candidato, o tema acima supracitado, já foi abordado em tópicos anteriores.

Linux

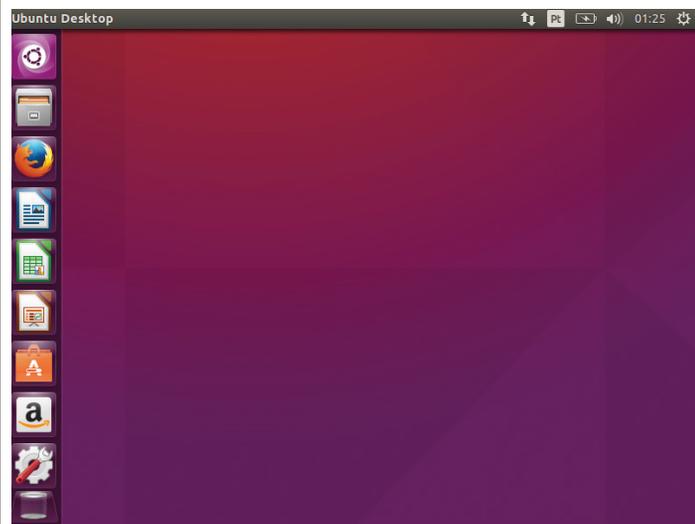
O Linux não é um ambiente gráfico como o Windows, mas podemos carregar um pacote para torná-lo gráfico assumindo assim uma interface semelhante ao Windows. Neste caso vamos carregar o pacote Gnome no Linux. Além disso estaremos também usando a distribuição Linux Ubuntu para demonstração, pois sabemos que o Linux possui várias distribuições para uso.



Linux Ubuntu em modo texto:

```
Ubuntu:~$ ls
Downloads      Music          Public         Videos
ts examples.desktop Pictures Templates
Ubuntu:~$
```

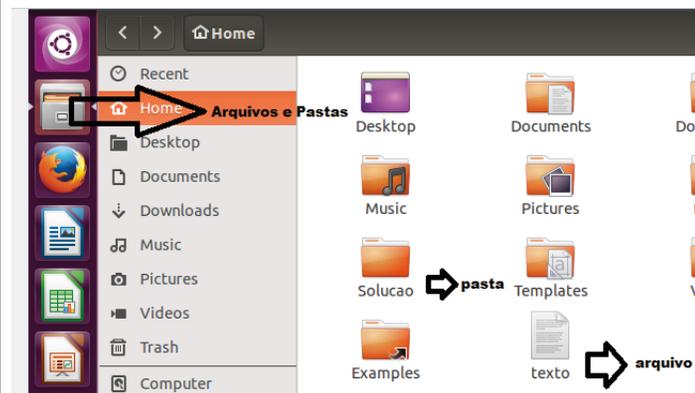
Linux Ubuntu em modo gráfico (Área de trabalho):



Conceito de pastas e diretórios

Pasta algumas vezes é chamada de diretório, mas o nome “pasta” ilustra melhor o conceito. Pastas servem para organizar, armazenar e organizar os arquivos. Estes arquivos podem ser documentos de forma geral (textos, fotos, vídeos, aplicativos diversos).

Dentro deste contexto temos uma hierarquia de pastas.



No caso da figura acima temos quatro pastas e quatro arquivos.

Arquivos e atalhos

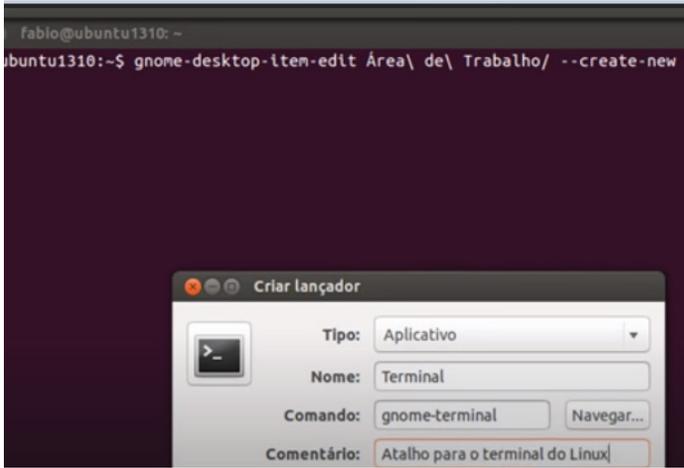
Como vimos anteriormente: pastas servem para organização, vemos que uma pasta pode conter outras pastas, arquivos e atalhos.

- Arquivo é um item único que contém um determinado dado.

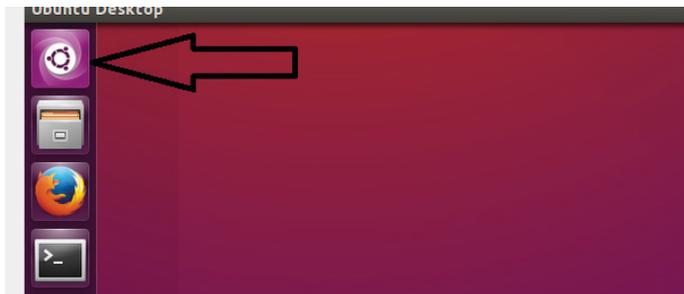
Estes arquivos podem ser documentos de forma geral (textos, fotos, vídeos e etc.), aplicativos diversos, etc.

• **Atalho** é um item que permite fácil acesso a uma determinada pasta ou arquivo propriamente dito.

No caso do Linux temos que criar um lançador que funciona como um atalho, isto é, ele vai chamar o item indicado.



Perceba que usamos um comando para criar um lançador, mas nosso objetivo aqui não é detalhar comandos, então a forma mais rápida de pesquisa de aplicativos, pastas e arquivos é através do botão:



Desta forma já vamos direto ao item desejado

Área de transferência

Perceba que usando a interface gráfica funciona da mesma forma que o Windows.

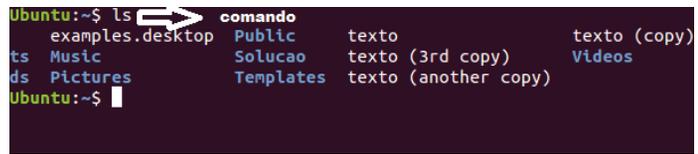
A área de transferência é muito importante e funciona em segundo plano. Ela funciona de forma temporária guardando vários tipos de itens, tais como arquivos, informações etc.

– Quando executamos comandos como “Copiar” ou “Ctrl + C”, estamos copiando dados para esta área intermediária.

– Quando executamos comandos como “Colar” ou “Ctrl + V”, estamos colando, isto é, estamos pegando o que está gravado na área de transferência.

Manipulação de arquivos e pastas

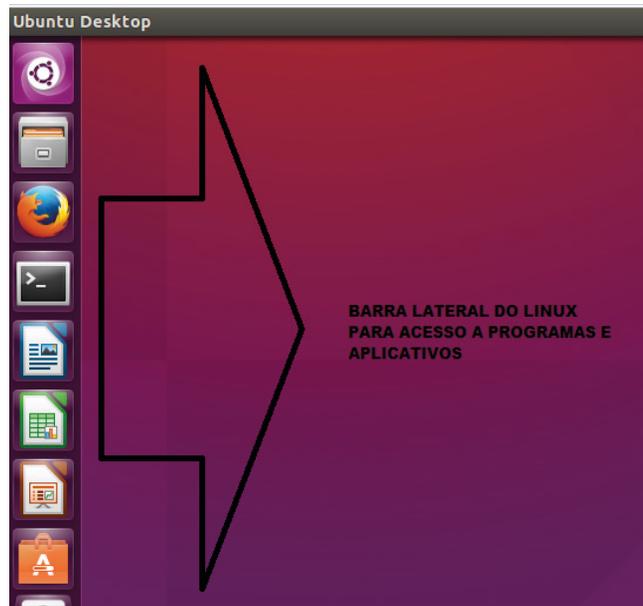
No caso da interface gráfica as funcionalidades são semelhantes ao Windows como foi dito no tópico acima. Entretanto, podemos usar linha de comando, pois já vimos que o Linux originalmente não foi concebido com interface gráfica.



Na figura acima utilizamos o comando **ls** e são listadas as pastas na cor azul e os arquivos na cor branca.

Uso dos menus

Como estamos vendo, para se ter acesso aos itens do Linux são necessários diversos comandos. Porém, se utilizarmos uma interface gráfica a ação fica mais intuitiva, visto que podemos utilizar o mouse como no Windows. Estamos utilizando para fins de aprendizado a interface gráfica “GNOME”, mas existem diversas disponíveis para serem utilizadas.



Programas e aplicativos

Dependendo da distribuição Linux escolhida, esta já vem com alguns aplicativos embutidos, por isso que cada distribuição tem um público alvo. O Linux em si é puro, mas podemos destacar duas bem comuns:

- Firefox (Navegador para internet);
- Pacote LibreOffice (Pacote de aplicativos semelhante ao Microsoft Office).

QUESTÕES

1. (CESPE BANCO DA AMAZÔNIA) O Linux, um sistema multitarefa e multiusuário, é disponível em várias distribuições, entre as quais, Debian, Ubuntu, Mandriva e Fedora.
 CERTO
 ERRADO

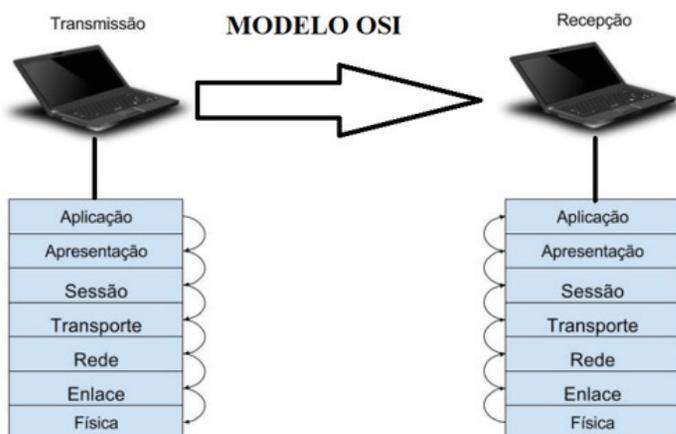
REDES DE COMPUTADORES E SEGURANÇA

CONCEITOS, FUNDAMENTOS, ARQUITETURA, MEIOS DE TRANSMISSÃO;

Uma rede é um ambiente que visa o acesso e o compartilhamento de recursos aumentando assim a colaboração entre seus membros. Estes membros colaboram entre si por meio de mensagens de áudio, vídeo e utilização de recursos estabelecendo assim uma efetiva comunicação.

Noções básicas de transmissão de dados

Para entendermos os conceitos, abaixo vamos analisar a figura:



A figura acima representa o modelo "OSI". Este modelo relata a transmissão de um conteúdo de um ponto A para um ponto B.

Percebemos que para sair da origem e chegar no destino, um pacote (conteúdo) passa por várias camadas que conferem e validam os dados.

Tipos de enlace

De acordo com a figura anterior vemos que existem camadas internas ao transmitirmos uma informação. A camada enlace é a responsável pela conferência dos dados e controle do fluxo de dados. Dentro deste contexto o enlace pode trabalhar de maneira ponto a ponto (computador para computador), ou broadcast (uso da rede para transmissão simultânea de dados), possibilitando assim o acesso ao meio.

Códigos

Os dados transmitidos pelo computador são codificados de tal forma que ele consiga trabalhar dentro do explicado.

Por exemplo, uma imagem pode ser representada da seguinte forma:

```
0 1000001 00100000 01001100 01001001 01000010
0 1000100 01001111 01010010 01000101 01010011
0 1001111 00100000 01001101 01000101 01001110
```

Desta forma os conceitos de conferência, correção, redundância, controle de requisição são executados na transmissão dos arquivos em geral (imagens, vídeos, e-mail, etc.).

Modos e meios de transmissão

Modos de transmissão: Existem três modos, conforme figura abaixo:

- **Modo Simplex:** Simplex é um modo de comunicação em que há um transmissor e um receptor da mensagem e esses papéis nunca se invertem no período da comunicação.

- **Modo Half-Duplex:** é um modo de comunicação em que há um transmissor e um receptor da mensagem e ambos podem enviar e receber mensagens, mas nunca simultaneamente

- **Modo Full-Duplex:** é um modo de comunicação em que há um transmissor e um receptor da mensagem e ambos podem enviar e receber mensagens simultaneamente.

Notamos que ao modo Full-Duplex é uma comunicação bidirecional, muito usada na sociedade por meio do Smartphones, vídeos conferências, etc.

Meio de transmissão é a utilização de uma forma de "transporte" da informação, no caso temos os guiados e os não guiados, conforme abaixo:

MEIO DE TRANSMISSÃO	DESCRIÇÃO
GUIADOS	São por meios de fios e cabos, pelos quais os dados são transmitidos via pulsos elétricos e entregues ao destino. São exemplos desta tecnologia cabos coaxiais, cabos par trançado, fibra óptica, etc.
NÃO GUIADO	São aqueles que não dependem de Fios ou cabos por exemplo: Wi-Fi, bluetooth, satélite, antenas, tecnologias 3G, 4G, 5G.

Camada física

A camada física na verdade diz respeito aos cabos, fibra óptica, roteador, repetidor, modem, conversores, etc. A camada física são os meios de conexão por onde irão trafegar os dados.

ANÁLISE E DESENVOLVIMENTO DE SISTEMAS

LÓGICA DE PROGRAMAÇÃO: ALGORITMOS, FLUXOGRAMAS, PROGRAMAÇÃO ORIENTADA A OBJETOS, DEPURAÇÃO.

Lógica de programação e algoritmos

Um algoritmo, na verdade, é uma receita que define uma tarefa passo a passo, posteriormente este algoritmo é convertido em uma linguagem de computador para que essa tarefa seja executada. Em termos mais técnicos, podemos dizer que um algoritmo é uma sequência finita e lógica que é definida por instruções que devem ser seguidas para resolver uma tarefa.

Como dissemos um algoritmo é uma definição de sequência de operações. No nosso dia a dia utilizamos algoritmos intuitivamente, por exemplo, vejamos abaixo:

Algoritmo para a troca de uma lâmpada

Início

- Verifica se o interruptor está desligado;
- Procura uma lâmpada nova;
- Pega uma escada;
- Leva a escada até o local;
- Posiciona a escada;
- Sobe os degraus;
- Retira a lâmpada queimada;
- Coloca a lâmpada nova;
- Desce a escada e aciona o interruptor;

Fim

Fluxograma

O fluxograma na verdade é a representação gráfica do algoritmo. O fluxograma utiliza-se de figuras que iniciam uma sequência de operações e finalizam esta sequência, demonstrando assim de forma lógica as operações.

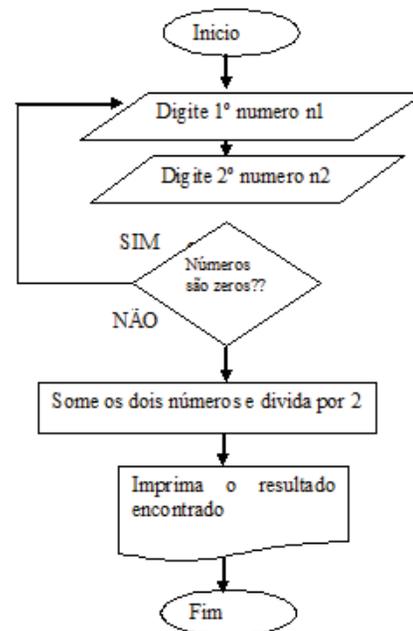
Simbologia:

Veremos a seguir cada símbolo e qual a sua função.

	Este símbolo é usado para o início e fim da sequência
	Este símbolo é utilizado para estabelecer a sequência de tarefas.
	Este símbolo é utilizado quando precisamos manipular dados: cálculos, atribuições, etc.

	Indica quando teremos que mudar o fluxo de operações de acordo com uma condição pré-estabelecida.
	O conector utilizado para indicar a continuidade do desenho na mesma página.
	É utilizado para representar a continuidade do desenho em outra página.
	É utilizado, quando há necessidade de uma entrada pelo usuário via teclado, isto é, uma entrada de dados.
	É utilizado quando será exibida alguma informação no monitor, isto é, uma saída de dados.

Exemplo de fluxograma:



O exemplo acima demonstra um fluxograma com entradas, saídas e processamento. Este fluxograma deverá ser escrito em uma linguagem de computador de acordo com estas sequências.

Programa orientada a objetos

Programação orientada a objetos é um paradigma de programação onde se tenta aproximar vida real e conceitos, paradigmas e outros artefatos de programas. Isso torna mais intuitiva a programação e mais organizada podendo se reaproveitar partes em outros projetos quando se fizer necessário.

Dentro deste paradigma a programação orientada a objetos trabalha com coisas materiais ou abstratas que podem ser percebidas pelos sentidos humanos e descritas por suas características, comportamentos e estados atuais.

Podemos citar alguns exemplos como objetos tratados na programação orientada a objetos: Carros, Pessoas, etc.

Percebemos que quando usamos este paradigma temos entidades que possuem características comuns e que podem ser readaptadas em outros contextos.

Algumas linguagens orientadas a objeto

- Java
- Python
- php

O que compõe um objeto?

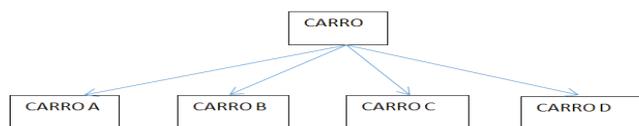
Atributos: Características do objeto, como nome, idade, endereço, telefone, etc.

Métodos: Ações de um objeto, como andar, escrever, nadar, etc.

O que é uma Classe?

Classe é um grupo que corresponde a itens comuns, desta forma podemos criar um objeto derivado de uma classe. Na figura abaixo temos a classe carro (PAI) e a partir daí temos quatro objetos carro (Filho)

Exemplo:



Os quatro pilares da programação orientada a objeto

Esses pilares são importantes nas definições de classes e objetos. Abaixo apresentamos os pilares:

• Abstração

É como vamos criar os objetos, na verdade os objetos são abstrações do mundo real, por isso devemos avaliar, pois os objetos deverão fazer sentido no mundo real para aplicarmos o conceito de abstração.

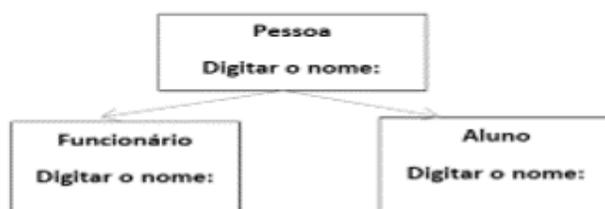
• Encapsulamento

Quando nós criamos um objeto (Filho) a partir de uma classe (Pai), não é uma cópia total da classe Pai, ao criarmos uma instância (cópia) gerando os objetos (Filhos), alguns detalhes ficam ocultos na geração, isto é, não ficam expostos. Daí o conceito de encapsulamento que protege o objeto.

O encapsulamento não é obrigatório, mas é uma boa prática de programação orientada a objetos.

• Herança

É um conceito fundamental na programação orientada a objetos. Herança é a criação dos objetos filhos com características comuns aos pais. Por exemplo: Se uma classe **Pessoa** tem um nome que deverá ser digitado, qualquer objeto criado a partir dela deverá herdar esta característica, de acordo com a figura abaixo:



• Poliformismo

Na verdade, poliformismo é a herança de características comuns, porém elas podem ser executadas de formas diferentes. Podemos dizer que poliformismo é uma herança que é diferenciada pela forma como ela é gerada.

Depuração

Depuração é um processo de conserto de algo que se sabe que está com erro no funcionamento. Em geral é pontual para um defeito e é um processo investigativo para descobrir a ocorrência de um erro.

Dentro deste contexto o programa é executado passo a passo e acompanhado com algumas paradas ocasionais, para investigar os resultados parciais e descobrir a causa do erro.

Geralmente o processo de depuração é executado por ferramentas específicas que permitem parar e avaliar parcialmente o processo, vendo o resultado parcialmente gerado.

ORGANIZAÇÃO DA INFORMAÇÃO: ESTRUTURAS DE DADOS, ARQUIVOS E REGISTROS, BANCOS DE DADOS.

Estruturas de dados

Estrutura de dados, de acordo com a ciência da computação, diz respeito a coleções de dados, valores e seus relacionamentos. É algo abstrato implementado de uma forma que pode ser trabalhado por meios computacionais.

Por sua vez, Algoritmos são sequências elaboradas que varrem uma estrutura de dados em busca da resolução de um determinado problema. Portanto é importante termos uma estrutura eficiente de acordo com o objetivo a ser atingido.

Dentro deste estudo, vamos notar que estes conceitos estão implícitos em toda teoria computacional e aplicado na tecnologia como um todo.

Vejamos o exemplo abaixo:

SÃO PAULO	MANAUS	VITÓRIA	RECIFE	RIO DE JANEIRO
-----------	--------	---------	--------	----------------

Este é um exemplo em que é guardada uma lista na memória do computador com algumas capitais do Brasil.

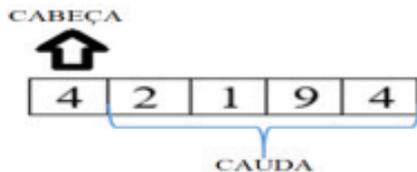
Estrutura de dados clássicas

Vetores

Os vetores, também conhecidos como arrays, são estruturas finitas compostas por um número fixo de elementos. Podemos dizer que um vetor é um compartimento com gavetas, fazendo uma analogia com algo concreto para entendemos.

Listas

A lista (sequência) é uma estrutura que os elementos seguem uma ordem, são armazenados e podem se repetir.



Perceba na figura cima que conseguimos identificar quem é o 1º elemento, o 2º elemento, e assim sucessivamente. Outro fato importante sobre listas é o conceito de “Cabeça e Cauda”, conforme apresentado na figura acima. Dentro deste contexto temos inúmeros paradigmas de programação.

Resumo sobre a estrutura lista

- Os elementos seguem uma ordem de armazenamento, sendo fácil identificar sua posição;
- Permitem a repetição de determinado elemento no decorrer da lista assim como os vetores;
- Cresce e diminui dinamicamente, diferentemente dos arrays, cujo tamanho é definido previamente;
- Facilidade de indexação e pesquisa, devido ao conceito de cabeça e calda;
- Uma lista tem uma estrutura linear, cada elemento está em um nó e aponta para o próximo com exceção do último nó, desta forma temos a pesquisa otimizada.

Filas

A fila é uma lista como mais especificidade. A fila é uma estrutura em que os elementos são sempre inseridos no início e removidos no final.

No caso da fila existe essa restrição ao inserir e remover elementos. É um conceito similar ao nosso convívio social. Vejamos no exemplo abaixo:

- Em uma fila no supermercado — entrar na fila, aguardar a vez e sair.



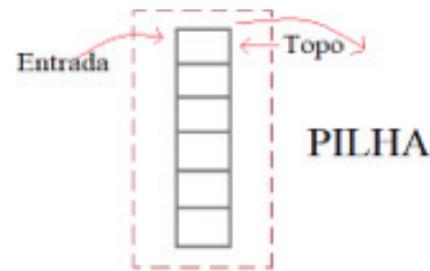
Podemos também usar o acrônimo “FIFO” (first in first out) para representar este processo.

Pilhas

Pilha é uma estrutura de dados abstrata linear com capacidade limitada que segue uma determinada ordem. Pilha é uma estrutura de dados simples pré-definida, que permite adicionar e remover elementos em uma determinada ordem. Quando um elemento é adicionado, este elemento vai para o topo da pilha, desta forma o elemento anterior é removido e este assume seu lugar.

O pedido pode ser LIFO (Last In First Out — Último a entrar, primeiro a sair) ou FILO (First In Last Out — Primeiro a entrar, último a sair).

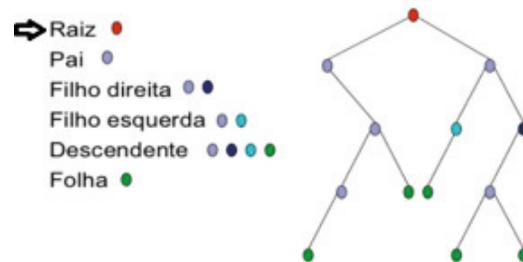
A figura abaixo, demonstra o funcionamento de uma pilha.



Árvore binária

Uma árvore binária é uma estrutura de dados caracterizada por nós, em que cada “nó pai” pode ter no máximo dois filhos, como demonstra as figuras abaixo:

Estrutura de uma árvore binária



LINGUAGENS DE BANCO DE DADOS

SQL. BANCO DE DADOS: FUNDAMENTOS DE GERÊNCIA DE BANCO DE DADOS RELACIONAL, CONCEITOS, COMPONENTES, FUNCIONAMENTO, SEGURANÇA E INTEGRIDADE, CÓDIGO ARMAZENADO (STORED PROCEDURES E TRIGGERS) (MY-SQL, ORACLE).

SQL

SQL é uma linguagem de Consulta de Dados com a qual podemos selecionar, deletar e apagar registros e executar outras funções no BD. É um padrão, que todos os bancos de dados utilizam, com algumas particularidades próprias relacionadas ao banco de dados em questão.

O SELECT é um dos principais comandos que são utilizados em SQL, pois com ele é possível selecionar e fazer consultas no banco de dados. Desta forma podemos pesquisar, por exemplo, todos os alunos cadastrados, através do comando abaixo:

SELECT * FROM alunos;

Ao ser executado, o comando acima irá retornar todos os alunos cadastrados no banco de dados.

Linguagem de Definição de dados

Existe também dentro da SQL os comandos DDL, que são responsáveis pela criação de objetos de definição de dados.

DDL ou Data Definition Language (Linguagem de Definição de dados) são os comandos que permitem ao usuário a criação de objetos e elementos associados.

CREATE DATABASE Escola	Cria o Banco de Dados Escola
ALTER TABLE Alunos ADD nome INT AFTER telefone	Altera a Tabela Alunos
DROP DATABASE Escola	Apaga o banco de dados Escola
DROP TABLE Alunos	Apaga a tabela Alunos

Linguagem de Manipulação de Dados

A linguagem DML, é responsável pela manipulação dos dados em si, por exemplo, temos uma tabela Alunos, responsável por manter todos os cadastros dos alunos.

Vejamos os comandos abaixo:

- INSERT

INSERT INTO aluno (id, nome, idade) VALUES (1, Jose da Silva, 30);

O comando acima irá criar o aluno Jose, com idade de 30 anos e ID 1.

- O UPDATE é utilizado para atualizar os dados de uma ou mais tabelas.

UPDATE aluno SET nome = Jose das Neves' WHERE id = 1;

Ao executar o comando acima, estaremos alterando o nome do usuário que possui o ID 1 para "Jose das Neves".

- DELETE, como seu próprio significado já diz, utilizaremos para excluir os dados de uma ou mais tabela em nosso banco de dados.

DELETE FROM usuario WHERE id = 1

Desta forma, ao executar o comando acima, estaremos excluindo o aluno que possui o ID 1 do nosso banco de dados.

MY-SQL

MYSQL é um sistema de gerenciamento de banco de dados relacional (SGBD) que utiliza a linguagem padrão SQL para a manipulação de seus dados. É um banco de dados multiplataforma que roda em Windows, Linux e Mac.

Vantagens do MySQL

- Open source;
- Gratuito;
- É um BD multiplataforma roda em Windows, Linux e Mac;
- Possui uma comunidade ativa e extensa;
- Pode usar nos mais variados tipos de aplicação;
- Facilidade de aprendizado e programação.

Tipos de dados no MySQL

- Dados Strings: char, varchar, text, entre outros;
- Dados numéricos: bit, smallint, int, float, entre outros;
- Dados data: datetime, date, time, year, entre outros.

Sabemos que o SQL é a linguagem padrão para os bancos de dados, desta forma podemos usar os comandos abaixo para trabalhar com o banco de dados.

CREATE DATABASE Empresa;	Criando o banco de dados Empresa
USE Empresa;	Colocando o banco de dados Empresa em uso
CREATE TABLE Funcionario (id INT, nome VARCHAR (255), PRIMARY KEY(id));	Criando a tabela Funcionario
INSERT INTO Funcionario (id, nome) VALUES (1, 'Maria');	Inserindo um registro de uma pessoa na tabela Funcionario
SELECT * FROM funcionario;	Mostrando todos os Funcionarios cadastrados
UPDATE usuario SET nome = 'Jose Silva' WHERE ID =1;	Atualizando o nome do Funcionario para Jose Silva cujo ID=1.

DELETE FROM funcionario WHERE id =3;	Apagando o Funcionario cujo ID=3
---	-------------------------------------

ORACLE

SQL é a linguagem padrão ANSI para trabalhar com bancos de dados relacionais. Desta forma ela é um padrão aceito pela indústria e trabalha com vários bancos de dados tais como: Oracle, MYSQL, Microsoft SQL Server, SQLITE, POSTGRESQL, e IBM DB2.

De acordo com a tabela abaixo, apresentamos as principais funções das instruções SQL.

INSTRUÇÃO SQL	FUNÇÃO
SELECT	Recuperação de dados.
INSERT UPDATE DELETE	Manipulação de dados.
CREATE ALTER DROP RENAME TRUNCATE	Definição de objetos.
COMMIT ROLLBACK SAVEPOINT	Controle de transações.
GRANT REVOKE	Controle de acesso.

A Linguagem SQL é uma linguagem declarativa, daí a ORACLE criou a PLSQL que é uma linguagem mais robusta que na verdade é uma extensão da linguagem padrão SQL. Sendo assim a PLSQL é um padrão exclusivo da ORACLE.

Diferenças entre as Linguagens SQL e PL/SQL

SQL	PL/SQL
É uma linguagem declarativa. Não é possível criar programas inteiros em SQL, por falta de instruções de controle de fluxo.	É uma linguagem imperativa. Possibilita a criação de programas completos, devido as instruções de controle de fluxo, etc.
Por ser uma linguagem padrão ANSI Pode ser usada em qualquer SGBD relacional.	Linguagem própria. Exclusiva apenas pela Oracle.

Banco de dados

Banco de Dados ou Base de Dados é a organização de um determinado assunto, isto é, a separação das entidades e seus relacionamentos. Dentro deste conceito temos os SGBD (Sistema de Gerenciamento de Banco de Dados), que permitem o controle total do Banco (Segurança, armazenamento, gestão, etc.)

Vejam os um exemplo do banco de dados NorthWind, que foi disponibilizado pela Microsoft para fins de estudo. Este banco de dados trata de uma loja comercial.

De acordo com a análise do banco, foram separadas as tabelas (entidades) abaixo:

Tabelas

- + [ícone] dbo.Categories
- + [ícone] dbo.Contacts
- + [ícone] dbo.CustomerCustomerDemo
- + [ícone] dbo.CustomerDemographics
- + [ícone] dbo.Customers
- + [ícone] dbo.Employees
- + [ícone] dbo.EmployeeTerritories
- + [ícone] dbo.EMPRESAS
- + [ícone] dbo.Filiais
- + [ícone] dbo.Order Details
- + [ícone] dbo.Orders
- + [ícone] dbo.Products
- + [ícone] dbo.Region
- + [ícone] dbo.Shippers
- + [ícone] dbo.Suppliers
- + [ícone] dbo.Territories

Essas entidades são chamadas “tabelas” (Categorias, Clientes, Produtos, Pedidos, etc.)

Perceba que este banco de dados é focado na gestão de vendas, portanto através de ferramentas de análise podemos extrair informações operacionais e gerenciais.

Vejam os um exemplo da tabela de produtos:

	ProductID	ProductName	SupplierID	CategoryID	QuantityPerUnit	UnitPrice	UnitsInStock	UnitsOnOrder	ReorderLevel	Discontinued
1	1	Chai	1	1	10 boxes x 20 bags	18,00	39	0	10	0
2	2	Chang	1	1	24 - 12 oz bottles	19,00	17	40	25	0
3	3	Aniseed Syrup	1	2	12 - 550 ml bottles	10,00	13	70	25	0
4	4	Chef Anton's Cajun Seasoning	2	2	48 - 6 oz jars	22,00	53	0	0	0
5	5	Chef Anton's Gumbo Mix	2	2	36 boxes	21,35	0	0	0	1
6	6	Grandma's Boysenberry Spread	3	2	12 - 8 oz jars	25,00	120	0	25	0
7	7	Uncle Bob's Organic Dried Pears	3	7	12 - 1 lb pkgs.	30,00	15	0	10	0
8	8	Northwoods Cranberry Sauce	3	2	12 - 12 oz jars	40,00	6	0	0	0
9	9	Mishi Kobe Niku	4	6	18 - 500 g pkgs.	97,00	29	0	0	1
10	10	Ikura	4	8	12 - 200 ml jars	31,00	31	0	0	0
11	11	Queso Cabrales	5	4	1 kg pkg.	21,00	22	30	30	0
12	12	Queso Manchego La Pastora	5	4	10 - 500 g pkgs.	38,00	86	0	0	0
13	13	Konbu	6	8	2 kg box	6,00	24	0	5	0
14	14	Tofu	6	7	40 - 100 g pkgs.	23,25	35	0	0	0
15	15	Genen Shouyu	6	2	24 - 250 ml bottles	15,50	39	0	5	0
16	16	Pavlova	7	3	32 - 500 g boxes	17,45	29	0	10	0
17	17	Alice Mutton	7	6	20 - 1 kg tins	39,00	0	0	0	1
18	18	Camaron Tigres	7	8	16 kg pkg.	62,50	42	0	0	0
19	19	Teatime Chocolate Biscuits	8	3	10 boxes x 12 pie...	9,20	25	0	5	0

Perceba que dentro desta tabela temos várias colunas (campos), isso nos indica que podemos aplicar a mesma analogia do EXCEL para montagem das tabelas de um banco de dados.

Diagrama MER

O diagrama MER (Modelo entidade relacionamento) nada mais é que a relação entre as entidades. Por exemplo: Percebemos que N produtos podem pertencer a 1 categoria, segundo a figura do modelo “MER” abaixo:

Exemplo:

COMPUTADOR	INFORMÁTICA
Mouse	Informática
Teclado Dell	Informática

SERVIDORES WEB

APACHE.

O Apache Web Server foi criado em 1995 pelo funcionário Rob McCool, na época integrante da NCSA (National Center for Supercomputing Applications). É um servidor web muito popular, utilizado principalmente no mundo Linux.

Apache é um servidor Free sob a licença GNU, isto é, fornece gratuitamente todos os recursos de disponibilização de páginas Web, E-mail(s), etc.

O nome Apache faz referência a uma tribo de índios americanos dotada de grandes estratégias e resistência.

O servidor Web Apache é compatível com o protocolo HTTP. Funcionalidades	Modular por meio de API(s)
Plataformas	Windows / Linux / Unix / Mac OS
Hardware	Processadores 64 no mínimo
Segurança e criptografia.	Modúlo para camada SSL (Securiy Layer)
Metodologia de Processamento	Multiprocessamento (MPMs — Multi- Processing Modules).

TECNOLOGIAS WEB: AJAX, WEBSERVICES (XML, RSS), CSS, DHTML

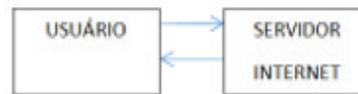
AJAX

AJAX (Asynchronous Javascript and XML) é uma comunicação assíncrona, isto é, a comunicação ocorre em segundo plano, sem percepção por parte do usuário, simplesmente o usuário vê a resposta, não houve espera, carregamento, etc., do lado do cliente (usuário).

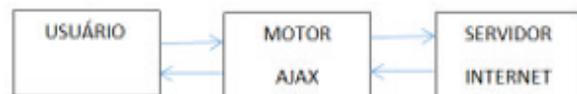
Hoje temos exemplos de aplicações que fazem o uso de requisições AJAX: são as notificações, twitters, chats, etc.

Vejamos a figura abaixo:

ABORDAGEM TRADICIONAL



ABORDAGEM AJAX



Como o AJAX funciona?

O AJAX funciona como um intermediário entre o cliente e o servidor: O cliente envia as solicitações, que são gerenciadas em segundo plano, otimizando assim a velocidade, a interatividade e a usabilidade das aplicações

WEBSERVICES (XML, RSS)

Os webservice como RSS são mecanismos de distribuição e atualização de milhares de sites ao redor do mundo.

RSS permite rápidas visualizações de notícias e atualizações.

EXEMPLO DE DOCUMENT RSS

```

<?xml version="1.0" encoding="UTF-8" ?>
<rss version="2.0">
<channel>
<title>Página teste</title>
<link>https://www.teste.com</link>
<description>isto é um teste</description>
<item>
<title>RSS Tutorial</title>
<link> https://www.teste.com xml/xml_rssteste.asp</link>
<description>RSS Exemplo</description>
</item>
<item>
<title> RSS Exemplo </title>
<link> https://www.teste.com /xml</link>
<description> https://www.teste.com </description>
</item>
</channel>
</rss>
  
```

Por que usar RSS?

- RSS foi projetado para mostrar somente os dados selecionados;
- Consumo de tempo menor, evitando os usuários fazer varreduras em busca de notícias;
- Rapidez de carregamento;
- Compartilhamento automático de informações;
- RSS permite a seleção de notícias, desta forma somente notícias relevantes relacionadas ao contexto poderão ser vistas.

Quem pode usar o RSS

- Webmasters que atualizam frequentemente seu site.
- Sites de notícias - Lista notícias com título, data e descrições
- Empresas - Lista para atualizar as notícias e novos produtos
- Calendários - Lista para atualizar os próximos eventos e dias importantes
- Calendários - Lista para atualizar os próximos eventos e dias importantes

Como funciona o RSS

- RSS é usado para compartilhar conteúdo dentre os sites.
- Com RSS, você registra seu conteúdo em empresas chamadas agregadoras.
- Para iniciar o processo de criação e compartilhamento de RDS, cria-se um documento e salva-se com XML e registra-se em um agregador. A partir daí o agregador faz a distribuição para os clientes.

CSS

O Cascading Style Sheets (CSS) é uma “folha de estilo” composta por “camadas” e utilizada para estilizar a apresentação (aparência) de sites da internet escritos em linguagens de marcação, tais como: (HTML, XML, e XHTML).

O CSS define como serão mostrados os elementos em uma página da internet, isso permite ao desenvolvedor separar o código do conteúdo do documento e do formato do documento.

Foi adotada para estilizar a apresentação de páginas na internet deixando-as mais atrativas visualmente aos usuários. Essa necessidade, fez com que páginas simples de HTML destinadas a apresentação de conteúdos incorporassem recursos do CSS para formatar elementos como layout, formatos, cores etc.

A boa prática diz que deverão ser mantidos separados os códigos CSS e HTML, para apenas invocá-los quando for preciso, evitando a repetição.

Exemplos de CSS

1. Conforme abaixo, o selector representa qualquer tag HTML. Dessa forma, podemos formatar os elementos internos de uma só vez. Ao invés de formatar todos os títulos H1 individualmente, usamos o código CSS para formatar todos de uma só vez. Desta forma sempre que a tag H1 que for utilizada no documento HTML, será formatada com a fonte Arial conforme especificado no CSS.

```
Selector {
    property:value;
}
h1 {
    font-family:Arial;
}
```

2. Conforme abaixo toda tag p do html será formatada com a cor azul:

```
p {
    color :blue;
}
```

3. Conforme abaixo toda a tag span do html será formatado com a cor verde e em negrito:

```
span {
    color :green;
    font-weight:bold;
}
```

4. O ID é um identificador único, neste caso a TAG é referenciada como “#unica”

Toda a TAG é referenciada com o símbolo “#”. O ID significa uma formatação exclusiva para uso. Cada página HTML pode ter somente uma única tag usando o ID definido.

```
#unica {
    /* Use as cores em notação hexadecimal para obter mais variedades. */
    background-color :#FFD700;
    color :#000000;
    font-family:Courier ;
}
```

5. Neste caso a formatação, está sendo na TAG SPAN dentro da classe listas.

Portanto ao referenciar a classe .listas a formatação da cor será obtida.

```
.listas span {
    font-size:2em;
    color :red;
}
```

6. Neste caso a formatação, está sendo na classe .primeira-div. Portanto ao referenciar a classe .listas a formatação será obtida.

```
.primeira-div {
    background-color :blue;
    /* Largura */
    width:150px;
    /* Altura */
    height:150px;
}
```

7. A formatação de links segue os padrões abaixo:

- link = link normal, não visitado;
- hover = quando o mouse está sobre o link;
- visited = um link que o usuário já visitou;
- active = link no momento em que é clicado.

```
#link-principal:link {
```

8. O parâmetro text-decoration é útil para indicar qual o tipo de sublinhado desejado (abaixo, acima, cortado ou nenhum)

```
Text-decoration:none;
font-size:20px;
font-family:Courier, sans-serif ;
background-color :blue;
color :white;
font-weight:bold;
```

9. O parâmetro text-transform útil para indicar qual o tipo de texto desejado maiúsculo ou minúsculo.

LINGUAGENS DE PROGRAMAÇÃO PARA WEB

PHP, JAVASCRIPT, HTML

PHP

PHP é uma linguagem de script utilizada no desenvolvimento de sites, para aumentar a interatividade e usabilidade do usuário. O processamento do código PHP se dá no servidor, do qual é devolvida uma resposta ao cliente que está acessando o site. Desta forma o código PHP é mesclado ao código HTML trazendo ao usuário uma melhor experiência e acesso a funções mais específicas que vão além do HTML.

CARACTERÍSTICAS DO PHP

Simplicidade	PHP tem uma sintaxe simples, similar a outras linguagens populares no mercado. Pode ser utilizada em vários paradigmas tais como: Procedural e programação orientada a objeto, de acordo com o conceito do projeto.
Eficiência	Excelente desempenho com integração com diversos banco de dados, tais como: SQL Server, MySQL, MariaDB, Oracle e SQLite.
Multiplataforma	As aplicações escritas em PHP podem ser desenvolvidas e funcionar em Windows, Linux, UNIX e MacOS, além de serem hospedadas em servidores na nuvem, para serem utilizadas por diversos navegadores.
Código aberto	PHP, permite que se faça o download e inicie o trabalho sem custos para o desenvolvedor.

EXEMPLO DE CÓDIGO HTML

<html>
<head>
<title> Como funciona o PHP em uma página HTML</title>
</head>
<body>
<?php echo "<p>Vamos programar em PHP?</p>";
/* Exemplo de um parágrafo em PHP */
?>
</body>
</html>

O que é possível fazer com PHP

— Desenvolvimento de sites dinâmicos: Sites dinâmicos são aqueles em que existe uma grande interatividade entre o cliente e o servidor, PHP permite que a resposta seja devolvida, evitando atualizações do usuário para obter respostas do servidor.

— Criação de plugins: Plugins na verdade são uma extensão, que podem ser adicionadas em outro programa para conceder mais recursos. PHP se torna ideal devido às suas características.

— Desenvolvimento de aplicações para a internet: As aplicações via Internet hoje são uma realidade, podemos utilizar recursos que vão além de um site para consulta.

JavaScript

É uma linguagem usada na internet para oferecer interatividade de nos sites.

Por exemplo: Ao digitarmos o CPF em um site, o SITE valida este número, emitindo uma mensagem ao usuário se o número está correto ou não. Esse tipo de atividade, cálculos e outras interações são executadas por meio da linguagem JavaScript.

Vejam os a seguir as características da linguagem:

— Exerce controle sobre o HTML e o CSS **para manipular comportamentos de páginas web;**

— **É independente de plataforma;**

— Permite a criação de funcionalidades para sites que não podem ser criadas com HTML e CSS;

— O código escrito em Javascript é embutido no próprio arquivo HTML (basta que o programador utilize a tag <script> no momento de escrever as linhas em Javascript;

— Utiliza uma sintaxe familiar, para quem já utiliza Java, C, ou C++;

— Seu modelo de objetos é baseado em protótipos (e não em classes, como é o caso do Java);

— Suporta funções sem requisições especiais de declaração;

— **É um tipo de programação dirigida por eventos: possibilita a criação de trechos de programa que respondem a eventos específicos, tais como um clique em um botão.**

Na figura abaixo temos um fragmento de um código JavaScript demonstrando esses conceitos, exibindo a mensagem "Olá Mundo".