

Respostas dos Exercícios

Java para Games **O tutorial para sua grande aventura**

Helder Henrique do Nascimento Peres



editora
VIENA

1ª Edição
Bauru/SP
Editora Viena
2015

Capítulo 2

1. Qual é o nome correto de uma letra em programação? E um conjunto de letras?

Caractere ou Character. String.

2. Qual é o nome dado a um valor numérico utilizado para fazer operações matemáticas?

Integer, Inteiro ou int.

3. O que acontece se eu somar um valor Inteiro com um valor String?

Haverá concatenação, ou seja, o valor numérico será somado com a String se tornando parte dela.

4. O que faz um programa de computador?

Le e manipula símbolos, trocando-os ou modificando-os.

5. Ao se criar a seguinte variável em um programa, “Int x” qual é o nome da variável e qual é o seu tipo?

O nome da variável é x e o seu tipo é int (ou inteiro, integer)

6. O que esses símbolos matemáticos (+, -, *, /, =, == e !=) significam em programação?

Adição, subtração, multiplicação, atribuição, igualdade e diferente ou não igual.

7. O que os seguintes símbolos lógicos significam (&&, ||, <, >)?

E lógico, ou lógico, menor, maior.

8. Quais são os dois principais comandos para laços de repetição?

Para e enquanto ou for e while.

9. Considere a seguinte situação:

```
Int a;
a=10;
se a <15 então:
  escreva "A é menor que quinze, cara!"
fim-se;
```

10. Sabemos que a variável do tipo inteiro chamada "a" vale 10. O comando escreva vai ser executado se a condição for verdadeira. Se "a" vale 10 a condição é verdadeira? Porque?

Sim, pois a condição diz que a tem que ser menor que quinze e "a" vale 10 e 10 é menor que quinze.

11. Um vetor contém apenas uma dimensão quanto a matriz pode ter duas ou mais. Essa frase é verdadeira?

Sim.

12. Considere a seguinte função:

```
int função_somar (int x, int y){
  int resultado;
  resultado = x+y;
  return resultado;
}
```

a) Qual é o tipo de retorno da função?
Int ou inteiro

b) Qual é o nome da função?
Função_somar

c) Quais são os parâmetros da função?
Int x e int y.

d) Qual é o retorno da função?
A variável resultado.

12. Usando seus conhecimentos, crie um algoritmo para trocar de camisa, veja o exemplo: Pegar sapato; Pegar meia; Colocar meia; Colocar sapato;

Tirar camisa velha;
Pegar camisa nova;
Colocar camisa nova;

13. Usando seus conhecimentos, crie um algoritmo para escovar os dentes.

Entrar no banheiro;
Abrir armário;
Pegar escova && passar pasta de dente;
Enquanto dentes!= limpo:
Escovar dentes;
Fim;

14. Usando seus conhecimentos, crie um algoritmo para trocar o pneu do carro.

Pegar macaco;
Usar macaco para levantar carro;
Pegar chave;
Retirar pneu_furado;
Colocar pneu novo;
Tirar descer carro && tirar macaco;

15. Crie um algoritmo para comprar ovos. Se a dúzia do ovo for menor que 5 reais, compre duas dúzias. Não esqueça de levar uma ou mais bandejas, caso o ovo esteja barato.

Pegar bandejas;
Ir ao mercado;
Se preço do ovo menor que 5 reais então:
Pegar duas bandejas;
Comprar duas dúzias de ovos;
Se não:
Comprar uma dúzia de ovo;

16. Crie um algoritmo para jogar vídeo game. Considere que você só tem a televisão e que o vídeo game está em uma caixa. Se você não tiver nenhum jogo em sua casa, você terá de ir à casa de algum amigo buscar.

```
Tirar vídeo game da caixa;
Instalar vídeo game na TV;
Verificar jogos;
Se não tiver jogos;
Ir na casa de um amigo;
Pedir jogo emprestado;
Se tiver jogo:
Colocar jogo no vídeo game;
Jogar;
```

17. Crie um algoritmo usando as variáveis do tipo inteiro e booleana para cortar gramas e funções. Não esqueça de declarar as funções no algoritmo (explicar como elas funcionam)Veja um exemplo:

```
Boolean cabelo_bagunçado;
Se cabelo_bagunçado==true então:
    Int pente;
    Pente= 20;
Boolean Pentear_cabelo(pente);

Se não:
Void Passar_perfume();

Função boolean pentear_cabelo(pente){

Pente- -;
Se pente <=0 então:
    Cabelo_bagunçado==false;

Return cabelo_bagunçado;
}

Função void passar_perfume()
{
    Int perfume;
    Perfume=perfume-10;

Return perfume;
}
```

```

int cortador=100;
Boolean grama_cortada=false;

Grama_cortada= Boolean Cortar_grama(cortador, grama_cortada);

Função Boolean cortar grama (cortador, grama_cortada){

Enquanto cortador>0 :
    Cortador--;

Se cortador<=0 :
    Grama_cortada=true;

Return grama_cortada;
}

```

- 18.** Crie uma variável Y e faça um laço de repetição que escreva cem vezes a frase: “Estude muito!”. Veja o exemplo:

```

int x;
para x de 0 até 1000 faça:
    escreva “frase maneira.”
fim-para

```

```

int y;
para y de 0 até 99 faça:
    escreva “estude muito!”
fim-para

```

- 19.** Crie um laço de repetição para escrever 200 vezes a frase “Vamos nessa!”

```

int y;
para y de 0 até 199 faça:
    escreva “vamos nessa!”
fim-para

```

- 20.** Crie um laço de repetição para escrever 1000 vezes a frase “É hora da força do tempo!”

```
int y;
para y de 0 até 999 faça:
    escreva “é hora da força do tempo!”
fim-para
```

- 21.** Quantas vezes o laço de repetição escreverá a frase “Vai digiovo!”

```
Int d;
para d de 0 até 100 faça:
    escreva “vai digiovo!”
fim-para
```

cento e uma vezes.

- 22.** Analise o seguinte comando:

```
Int a;
String b;

b=”Teste”;

enquanto b == “teste” faça:
    escreva a++;
fim-enquanto;
```

O que acontecerá neste comando? Explique.

Acontecerá um loop infinito, pois a condição não deixará de ser verdadeira.

- 23.** Analise o seguinte comando:

```
Int a=100;

enquanto a >0 faça:
    escreva a--;
fim-enquanto;
```

O que acontecerá neste comando? Explique

Enquanto a for maior que zero a será escrito na tela, de maneira regressiva.

Capítulo 3

1. Eiffel, Java, C#, Python e C++ são linguagens estruturadas?

Não, são linguagens orientadas a objetos.

2. Qual é a vantagem de uma linguagem orientada a objetos?

Ela permite modelar objetos em um programa para que ele se corresponda a um objeto da vida real como uma pessoa, máquina, animal, comida e etc.

3. Um objeto é tudo aquilo que possui características e ações. Essa frase está correta?

Sim.

4. Como se chama as características e ações de um objeto em JAVA?

Atributos e métodos.

5. O que é uma classe?

É uma definição de como será o objeto.

6. Se uma classe chamada “Carro” criar um objeto, esse objeto pertencerá a classe carro?

Sim.

7. Todos os atributos e métodos definidos em uma classe, serão instanciados em seus objetos?

Sim.

8. O que é o método construtor de uma classe?

Um métodos que cria objetos e define os valores dos mesmos.

9. É possível implementar um ou mais construtores personalizados para definir diferentes valores ao se instanciar um objeto?

É possível, cada construtor pode ter definições de valores do estado do objeto diferentes, ou seja, é possível criar objetos com valores diferentes usando construtores diferentes.

10. Qual é a grande utilidade do conceito de Herança?

Reaproveitamento de código.

11. Como funciona um conceito de herança?

Uma classe B (classe filho) herda todos os atributos e métodos de uma classe A (classe pai), tendo não somente os seus dados mas também os dados de sua classe pai.

12. A classe pai é a classe que recebe a herança e a classe filho é a que passa a herança. Essa frase é verdadeira?

Não, a classe pai é aquela que se passa a herança e a classe filho é aquela que recebe a herança.

13. Qual o comando em Java usado para indicar herança?

Implements

14. O que é uma sobrecarga em JAVA?

Quando um método possui o mesmo nome, mas parâmetros diferentes.

15. O que são os parâmetros de um método?

São variáveis que são visíveis somente no método e que receberão valores e que receberão valores externos para que esses valores sejam executados na função.

16. Dois métodos podem ter o mesmo nome?

Sim, desde que tenham parâmetros diferentes.

17. O que é uma restrição em JAVA?

Algo que impede (restringe) o uso de recursos de um objeto ou classe.

18. Quais são as restrições em JAVA?

Public, protected e private.

19. Um atributo marcado como publico (public) pode ser visto globalmente (ou seja, em todas as classes e pacotes do programa)?

Sim.

20. Um atributo marcado como privado (private) pode ser visto globalmente (em todas as classes e pacotes do programa)?

Não, apenas pode ser visto dentro de sua própria classe.

21. Um atributo marcado como protegido (protected) pode ser visto globalmente (em todas as classes e pacotes do programa)?

Não, somente dentro de sua própria classe, subclasses (classes filho) e seu próprio pacote.

Capítulo 4

1. Verdadeiro ou falso?

(V) Todo atributo em Java deve ser escrito com letra minúscula

(F) Toda classe deve ser escrita com letra minúscula e objetos com letra maiúscula

(V) Em objetos com nomes compostos o primeiro nome deve ser minúsculo e o segundo deve iniciar com letra maiúscula e sem espaço.

2. Crie uma classe no seu ambiente de programação.

```
public class Classe {
```

```
}
```

3. Crie uma sequencia de pacotes com o seguinte domínio “br.com.teste”

Isso dependerá o ambiente de programação utilizado. Mas geralmente os comandos são Novo pacote, br.com.teste

4. Quantos pacotes foram criados no exercício anterior? Quais são os seus nomes?

3 pacotes. Um chamado br, outro chamado com e outro chamado teste.

5. Escreva o seguinte código em seu ambiente de programação:

```
public class Classe {

    public void method (){

        int variavel;
        variavel =100;

        variavel=variavel+10;

        System.out.print(“O Resultado é “+variavel);

    }

}
```

a) Qual é o nome da classe acima?
Classe.

b) Qual é o tipo do método method?
Void

c) Qual é o nome da variável do tipo inteiro (ou int)?
Variável

d) Qual frase vai ser impressa no final do método?
“O Resultado é 110”

6. Copie o código em seu ambiente de programação e corrija o erro do seguinte código:

```
public class Classe {
```

}

```
public void method () {
```

```
int variavel;
```

```
variavel = 100;
```

```
variavel=variavel+10;
```

```
System.out.print("O Resultado é "+variavel);
```

}

O erro está em deixar todo corpo do método fora da classe.

7. Explique qual foi o erro do exercício anterior.

O erro está em deixar todo corpo do método fora da classe.

8. Onde fica o corpo da classe?

No abrir e fechar de chaves.

9. Todo o conteúdo da classe tem que estar dentro de seu corpo?

Sim.

- 10.** Crie uma segunda classe chamada *Teste*, no mesmo pacote onde foi criado a classe de nome *Classe* e instancie um objeto do tipo *Classe* dentro da classe *Teste*. De o nome desse objeto de *c*.

```
public class Teste {

    Classe c= new Classe();

}
```

- 11.** Qual o comando que representa herança em JAVA?

Extends.

- 12.** Crie uma sobrecarga com o método de nome “atividade”. O segundo método sobrecarregado deve ter dois parâmetros, um de nome “pergunta” sendo inteiro e outro de nome “resposta” sendo string.

```
public void atividade (){}

public void atividade (int pergunta, String resposta){}
```

- 13.** Considere o seguinte laço de repetição:

```
public void atividade (int pergunta, String resposta){

    int x;

    for (x=0;x<=10;x++){

        pergunta=pergunta+10;

    }

    System.out.println(“ A “ + resposta+” é na verdade “+pergunta); }
```

- a) Em que método o laço de repetição se encontra?
No método “atividade”

b) Qual tipo de laço de repetição é esse? Para ou enquanto?
Para.

c) Quantas vezes esse laço será executado?
11 vezes.

d) Crie o método principal em sua classe chamada Teste :
`public static void main(String[] args) {}`. Dentro deste método, execute o método atividade com os seguintes parâmetros: 0, “Desaparecido”. Qual vai ser o resultado da execução?
A Desaparecido é na verdade 110.

14. Você reparou que para poder executar alguma atividade é necessário criar um método especial. Qual é esse método?

O método principal, ou `public static void main`.

15. É possível executar qualquer operação em Java, fora do método principal?

Não.

16. Descreva uma caneta. Quais são suas características (atributos)? Quais são suas funções (métodos)?

Atributos: cor, espessura, tampa, marca, preço, modelo.

Métodos: escrever, rabiscar, pintar, desenhar.

17. Descreva um animal. Quais são suas características (atributos)? Quais são suas funções (métodos)?

Atributos: cor, pelo, escama, pena, numero de patas, espécie, habitat, sexo, nome científico, predador natural e etc.

Métodos: nascer, crescer, se reproduzir, comer, morrer, correr, saltar e etc.

18. Descreva um carro. Quais são suas características (atributos)? Quais são suas funções (métodos)?

Atributos: cor, quantidade de portas, marca, modelo, fabricante, esportivo, popular, ar condicionado, radio.

Métodos: andar, frear, correr, mudar de marcha e etc.

19. Agora , utilizando uma das resposta que você deu em um dos exercícios, 16,17,18 crie uma classe com os métodos e atributos que você listou.

20. Considere o seguinte método

```
public String condição (String cor){

    if(cor=="Azul")
    {System.out.println("Tudo azul!");}

    if(cor=="Vermelho")
    {System.out.println("Meu coração é vermelho!");}

    if(cor=="Preto")
    {System.out.println("Back in Black, baby.");}

    if(cor=="Rosa")
    {System.out.println("Ui, que chique bem.");}

    return cor;

}
```

- a) Qual frase será impressa se o meu parâmetro for "Azul"?
Tudo azul!
- b) Qual frase será impressa se o meu parâmetro for "Rosa"?
Ui, que chique bem.
- c) Se meu parâmetro for "Verde" qual condição será verdadeira? Qual frase será impressa?
Nenhuma condição será verdadeira e nenhuma frase será impressa.
- d) O método retornará a variável cor, mesmo que nenhuma condição seja verdadeira?
Sim.
- e) Supondo que eu adicione o seguinte comando antes do return :

```
else {

    System.out.println("Sem cores!");

}
```

O que aconteceria meu parâmetro fosse verde? Por quê?

A frase “Sem cores!” apareceria porque a ultima condição não foi verdadeira.

f) O comando else serve para todos os comando if ou apenas para aquele que vem antes dele?

Apenas para aquele que vem antes.

g) Se meu parâmetro fosse “Rosa”, o comando else seria ativado? Explique.

Não. Rosa é a condição verdadeira. Apenas entraria no bloco de comando else se Rosa não fosse verdadeira.

h) Se meu parâmetro fosse Vermelho, o comando else seria ativado? Quais frases apareceriam? Por quê?

Sim. “Meu coração é vermelho” e “Sem cores!”, pois, o else se refere apenas ao ultimo comando if que seria o Rosa. Sendo que a cor é Vermelho e não Rosa o else seria ativado sim.

21. É obrigatório se colocar um comando else após um comando if?

Não.

22. Como se identifica as posições de um vetor?

Por números representando o índice de cada posição.

23. Quando se cria um vetor ou matriz, qual é o seu primeiro índice?

0. Toda matriz e vetor começam no zero.

24. É possível ou impossível criar uma matriz com mais de duas dimensões?

É possível.

25. Comentários servem para ajudar a organizar o código. Isso é considerado uma boa prática? Cite outra função do comentário.

Sim. Serve para explicar o código e seus comandos.

Capítulo 5

1. Qual é o modelo mais aceito de banco de dados?

Modelo Entidade Relacional

2. Quem criou e quem aperfeiçoou e definiu o modelo de Entidade Relacionamento?

Edgar Frank Codd e Peter Chen

3. Quais são os três elementos que compõem o Modelo de Entidade Relacionamento?

Entidades, Atributos e Relacionamentos.

4. Qual é o nome que uma variável que guarda um ou mais caracteres (String) recebe em banco de dados?

Varchar

5. Qual o nome que do tipo de variável usada para guardar grandes arquivos indefinidos (de mídia, projetos e etc)

Blob.

6. Apesar de SQL ser uma linguagem unificada, existe diferença de uma plataforma para outra?

Sim, apesar de mínima.

7. Usando os conhecimentos adquiridos neste capítulo, configure um banco de dados utilizando mySql e crie um esquema ou banco de dados com o nome de mundo.

`create database mundo;`

8. Agora, dentro do banco de dados mundo, crie uma tabela chamada país, com os atributos , do tipo String chamados estado e município. Lembrando que não se usa acentuação em SQL

```
use database mundo;
create table pais(
id integer not null auto_increment primary key ,
municipio varchar(50) ,
estado varchar(50)
);
```

9. Para que serve uma chave primária?

Para definir unicamente um conjunto de valores (tupla) dentre de uma tabela. Serve como índice.

10. Uma chave primária pode ser nula e ter valores duplicados?

Não.

11. Para que serve uma chave estrangeira?

Para relacionar dados de uma ou mais tabelas distintas.

12. Utilizando seus conhecimentos, preencha os seguintes valores na tabela pais criada no exercício 8: (1,'São Paulo','SP') , (2,'Curitiba','PR'), (3, 'Londrina','PR'), (4,'Arapongas','PR'), (5,'Não-me-Toque','RS').

```
insert into pais values (1,'São Paulo','SP'),
insert into pais values (2,'Curitiba','PR'),
insert into pais values (3, 'Londrina','PR'),
insert into pais values (4,'Arapongas','PR'),
insert into pais values (5,'Não-me-Toque','RS');
```

13. Qual é o comando que se dá para confirmar uma ação em MySQL?

Commit;

14. Selecione agora, as cidades do estado do Paraná?

```
Select cidade from pais where estado = 'PR';
```

15. Que comando é usado para desfazer uma ação?

Rollback;

16. Crie uma tabela chamada Café com dois atributos tipos inteiros peso e porcentagem.

```
create table cafe(
  peso int,
  porcentagem int
);
```

17. Com suas palavras, o que o seguinte comando quer dizer insert into pais values (3, 'Londrina','PR');?

Que o valor 3, Londrina, PR está sendo inserido na tabela chamada pais.

18. O que o seguinte valor está escrito: insert into jogo values ('Sonic Battle');

Valor 'Sonic Battle' está sendo inserido na tabela jogo.

19. Considere a seguinte tabela:

Tabela: Pokemon
 Nome varchar,
 HP int,
 Ataque int,
 Defesa int,
 Ataque_Especial int,
 Defesa_Especial int,
 Velocidade int;

a) Se eu inserir os seguintes valores: “400, Wailord, 135, 300,285,146,110” oque acontecerá?Porque?

Haverá um erro porque se espera um valor do tipo varchar (string) como primeiro valor é passado um valor int como primeiro valor. Isso acontece porque na ordem em que é criado os atributos deve ser a mesma ordem para inserir os atributos.

b) Como seria a sequencia correta para inserir o valor?
 “Wailord, 400, 135, 300,285,146,110”

20. O que o seguinte comando quer dizer : `select * from PlayStation;`

Selecionará todos os valores da tabela PlayStation.

21. O que o seguinte comando significa : `select jogo from SegaSaturn;`

Que todos os valores do atributo jogo, da classe SegaSaturn serão selecionados.

22. O que o seguinte comando significa: `select * from Jogador where id=2;`

Selecione todos os dados do jogador cujo 1 é igual a dois

Capítulo 6

1. Ao se criar uma frame, qual é o conceito necessário para usarmos e fazê-la funcionar adequadamente?

O conceito de herança.

2. Utilizando seus conhecimentos, crie uma frame.

```

+ import java.awt.*;

public class NeoFrame extends Frame{

    public NeoFrame(){

        this.addWindowListener(new WindowAdapter(){
            public void windowClosing(WindowEvent we)
            {
                System.exit(0);
            }
        });
    }
}

```

3. Qual é o formato de imagem compatível com o Java?

- (X) JPEG
- () Bitmap
- (X) PNG
- () GIF
- () TFF

4. Por que o formato PNG é o mais adequado para se utilizar em jogos?

Porque ele remove o fundo da imagem.

5. Crie uma canvas em seu projeto

```
import java.awt.*;

class GraphicsProgram extends Canvas{

    public GraphicsProgram(){
        setSize(200, 200);
        setBackground(Color.white);
    }

    public static void main(String[] args){

        //GraphicsProgram é uma classe do tipo Canvas
        //já que herda de Canvas

        GraphicsProgram GP = new GraphicsProgram();

        //criamos uma nova frame... a NeoFrame
        NeoFrame f = new NeoFrame();
        f.setSize(800, 300);

        //E adicionamos nossa canvas para a frame.

        f.add(GP);

        f.setVisible(true);
        // Para entender lembre que canvas significa tela e frame moldura
        // sendo assim, pintamos coisas em uma tela e pinduramos em uma moldura
    }
}
```

6. Escreva o código necessário para habilitar o teclado em sua frame.

```
public class Teclado implements KeyListener {

    public void keyPressed(KeyEvent e) { // Verifica se a tecla é pressionada

        e.getKeyCode(); // pega o código da tecla
    }
}
```

7. O Método `getKeyCode()` retornará o valor da tecla pressionado. O que devemos fazer recuperar o valor gerado por ele? Que tipo de valor ele retornará?

Atribuir a uma variável. Um valor do tipo inteiro.

8. Como vamos saber qual valor numérico cada tecla tem?

Olhando a tabela do Java para cada tecla do teclado.

9. É possível retornar caracteres? Qual é o método?

`getKeyChar()`

10. Para fazermos nosso personagem se movimentar para esquerda ou direita o que devemos fazer?

Deve-se alterar sua variável correspondente ao eixo X da tela. Incrementando-a faz o personagem mover-se para frente (direita). Decrementando-a faz o personagem mover-se para trás (esquerda).

11. O que devemos fazer para nosso personagem se movimentar para cima ou para baixo?

Devemos alterar sua variável correspondente ao eixo Y. Incrementando-a ele irá para baixo, decrementando-a ele irá para cima.

12. Qual é o método que desenhara uma imagem na tela?

`g.drawImage()`

13. Considere o seguinte método:

```
g.drawImage(this.getSocoDir(a),this.getX(),this.getY(),
this.largura,this.altura,gp);
```

a) O que o primeiro parâmetro representa?

A imagem a ser desenhada na tela.

b) O que o segundo e o terceiro parâmetros representam?

A posição na tela onde a imagem será desenhada.

c) O que o quarto e quinto parâmetros representam?

A altura e a largura da imagem a ser desenhada.

d) O que o ultimo parâmetro representa?

O objeto da classe Canvas onde a imagem será desenhada.

14. Crie dois personagens. Em seguida, crie uma classe para cada um desses personagens. Depois, crie uma classe somente para checar a colisão desses personagens. Caso tenha alguma duvida de como fazer , olhe no código do jogo “Jhonny Adventure” deixado no capítulo 7. Sua missão final é criar um demo do jogo onde pelo menos haja dois personagens que se possam movimentar e colidir. Ah, em algum lugar da apostila há o código usado para se utilizar som, será que você consegue descobri-lo? Boa sorte!

15. Para que serve o seguinte código?

```
import java.util.TimerTask;
```

```
public class Temporizador extends TimerTask{
}
```

Para criar uma classe de temporizador.