Lista 25 – Teóricos

1. Analise o seguinte código Java e escreva nas linhas a saída gerada pela execução do programa.

```java
// Início do arquivo Ex1.java
package ex1;

public class Ex1 {

	public static void main(String[] args) {

		int x = 10;
		int repeticoes = 0;

		while(x > 0) {
			int y = calcula(x);
			x = y + 1;
			System.out.println(x);
			repeticoes++;
		}

		System.out.println(repeticoes + " repeticoes.");
	}

	private static int calcula(int x) {
		if(x > 0) {
			x = x - 3;
			return x;
		}
		return 0;
	}
}
// Final do arquivo Ex1.java
```

Saída:

_____
_____
_____
_____
_____
_____
_____
_____

2. Analise o seguinte código Java e escreva nas linhas a saída gerada pela execução do programa.

```
// Início do arquivo Ex2.java
package ex2;

public class Ex2 {

	public static void main(String[] args) {
		Cidade c = new Cidade(10000);
		c.incrementaPopulacao(c.getPopulacao() / 2);
		System.out.println(c.getPopulacao());
		c.incrementaPopulacao(c.getPopulacao() / 5);
		System.out.println(c.getPopulacao());
		c.incrementaPopulacao(c.getPopulacao() / 9);
		System.out.println(c.getPopulacao());
	}
}
// Final do arquivo Ex2.java


// Inicio do arquivo Cidade.java
package ex2;

public class Cidade {
	private int populacao;

	public Cidade(int populacao) {
		this.populacao = populacao;
	}

	public int getPopulacao() {
		return populacao;
	}

	public void incrementaPopulacao(int valor) {
		this.populacao += valor;
	}
}
// Final do arquivo Cidade.java
```

Saída:

_____
_____
_____
_____
_____

3. Analise o seguinte código Java e escreva nas linhas a saída gerada pela execução do programa.

```java
// Início do arquivo Ex3.java
package ex3;

public class Ex3 {
	public static void main(String[] args) {
		Piloto p1 = new Piloto("Senna","F1");
		p1.imprimir();
		Piloto p2 = new Piloto("Valentino Rossi","MotoGP");
		p2.imprimir();
	}
}
// Final do arquivo Ex3.java


// Início do arquivo Pessoa.java
package ex3;

public class Pessoa {
	private String nome;

	public Pessoa(String nome) {
		this.nome = nome;
	}
	public String getNome() {
		return nome;
	}
	public void setNome(String nome) {
		this.nome = nome;
	}
	public void imprimir() {
		System.out.print(nome);
	}
}
// Final do arquivo Pessoa.java
```

```java
// Início do arquivo Piloto.java
package ex3;

public class Piloto extends Pessoa {
	private String modalidade;

	public Piloto(String nome, String modalidade) {
		super(nome);
		this.modalidade = modalidade;
	}
	public void imprimir() {
		super.imprimir();
		System.out.println(" é da modalidade " + modalidade + ".");
	}
}
// Final do arquivo Piloto.java
```

Saída:

_____
_____
_____

4. Analise o seguinte código e escreva nas linhas a saída gerada pela execução do programa.

```java
// Início do arquivo Exercicio1.java
package exercicio1;

public class Exercicio1 {
	public static void main(String[] args) {
		Heroi c = new Heroi("Batman");
		c.atacar(1);
		c.atacar(2);
		c.atacar(1);
	}
}
// Final do arquivo Exercicio1.java

// Início do arquivo Personagem.java
package exercicio1;

public class Personagem {
	private String nome;

	public Personagem(String nome) {
		this.nome = nome;
	}
	public String getNome() {
		return this.nome;
	}
}
```

```java
// Final do arquivo Personagem.java
// Início do arquivo Heroi.java
package exercicio1;

public class Heroi extends Personagem {
	private int danoAdversario = 0;
	private boolean adversarioAbatido = false;

	public Heroi(String nome) {
		super(nome);
	}

	public void atacar(int golpe) {
	    switch(golpe) {
		    case 1 : {
			    System.out.println(getNome() + " aplica golpe " + golpe + "!");
			    danoAdversario = danoAdversario + (golpe * 2);
			    break;
		    }
		    case 2 : {
			    System.out.println(getNome() + " aplica golpe " + golpe + "!");
			    danoAdversario = danoAdversario + (golpe * 4);
			    break;
		    }
	    }

	    if(danoAdversario > 5) {
		    if(adversarioAbatido) {
			    System.out.println(getNome() + " sem nocao!");
		    } else {
			    System.out.println("Oponente fora de combate!");
			    adversarioAbatido = true;
		    }
	    }
	}
}  // Final do arquivo Heroi.java
```

Saída:

_____
_____
_____
_____
_____
_____
_____

5. Qual é a diferença entre uma classe e um objeto?

6. Qual é a diferença entre o modificador de acesso public e o private?

7. Explique o mecanismo de herança.

8. Explique a sobrescrita de método.

9. Dê um exemplo de sobrescrita de método.

10. Explique a sobrecarga de método.

11. Dê um exemplo de sobrecarga de método.

12. Marque V para verdadeiro e F para falsa. Justifique as afirmativas que forem falsas.

( ) Todas variáveis devem receber um tipo quando forem declaradas.
( ) O Java considera que as variáveis **number** e **Number** são idênticas.
( ) O método **Integer.parseInt** converte um inteiro em uma String.
( ) A expressão ((x>y) && (a<b)) é verdadeira se (x>y) for verdadeira ou se (a<b) for verdadeira.
( ) O uso do **break** não é obrigatório na estrutura de seleção **switch**.
( ) Supondo que **r** é um objeto de **Random**, **r.nextInt(10)** gera um inteiro aleatório entre 0 e 10.
( ) Para construir uma interface gráfica, podemos inserir um objeto JFrame em um objeto JPanel.
( ) Um método get pode ser usado para recuperar valores de dados **private** de um objeto.
( ) Os membros de classe **protected** são acessíveis somente por métodos da própria classe.
( ) Se a classe Alpha herda da classe Beta, a classe Alpha é a subclasse e Beta é a superclasse.
( ) A herança permite a reutilização de código.
( ) A superclasse, em geral, representa um número maior de objetos que sua classe.
( ) A subclasse, em geral, encapsula menos funcionalidades que a sua superclasse.
( ) A maneira correta de comparar duas Strings é por meio do operador ==.
( ) É possível ter uma sobrescrita de método sem uma herança.
( ) É possível ter uma sobrecarga de método sem uma herança.