

1. Corrija as seguintes afirmações:

Considere uma classe chamada Avaliacao. Para compilarmos e executarmos a classe devemos executar os seguintes comandos:

```
javac Avaliacao
```

```
javac Avaliacao.java
```

Resposta:

Javac Avaliacao.java

Javac Avaliacao

2. Qual o resultado da compilação do código abaixo e justifique sua resposta:

```
public class Aplicacao {  
  
    public static void main {  
  
        int contador;  
  
        System.out.println(contador++);  
  
    }  
  
}
```

Resposta:

```
public class Aplicacao {  
  
    public static void main(String[]args) {  
  
        int contador = 1;  
  
        System.out.println(contador++);  
  
    }  
  
}
```

O código estava errado antes pois não foi definido os parâmetros do main que é fundamental na execução dele além do contador não ter sido iniciado.

3. Explique detalhadamente o resultado após a execução abaixo do código:

```
public class Exemplo {  
  
    public static void main(String[] args) {  
  
        int a = 9;  
  
        int b = 0;  
  
        if ((a > 11) && (++b <= 100)){  
  
            System.out.println(a*b);  
  
        } else {  
  
            System.out.println(a-b);  
  
        }  
  
        }  
  
    }
```

Resposta:

Ao ser executado, o programa passa pela condicional do if so que com um curto circuito no valor de “a” pois esse já se tornar falso que por ser uma operação logica de multiplicação a operação toda retorna false ou seja o resultado da segunda condição não vai ser considerado e b não receberá um incremento ou seja o que vai ser executado é o else que retornará o resultado de 9 – 0.

4. Explique detalhadamente qual a saída da execução do trecho de código abaixo e justifique:

```
int op = 1;  
  
switch (op) {  
  
    case 1: System.out.println(op);  
  
    case 2: System.out.println(op);  
  
    default: System.out.println(op);
```

}

Resposta:

O código dessa questão tem como resultado 3 números 1 pois “op “como recebe o número um ele vai obrigatoriamente executar o primeiro caso porem ao final da execução do case 1 o programa não recebe a instrução de break ou seja todo código que for executável após o case 1 será executado até que algum break seja colocado no switch.

9. Compare as implementações da questão 6 e com a implementação das questões

7 e 8. Qual é a abordagem mais "estruturada" e qual a versão mais Orientada a

Objetos? Qual a mais modularizada e reutilizável? Entretanto, qual a mais

burocrática devido ao problema a ser resolvido?

Resposta: A questão mais estruturada é a 6 e as mais orientadas a objetos são a 7 e 8. A mais modularizada e reutilizável é a 7 e 8. As mais burocráticas são a 7 e 8.