

Aluna: Analu Sorbara

Responder às seguintes questões e entregar no Moodle em arquivo PDF.

1. Como o uso de várias threads Java permite escrever programas mais eficientes?

Permite que tarefas sejam executadas paralelamente, com isso, o programa consegue adiantar tarefas que normalmente estão aguardando execução em um código sequencial.

2. O uso de várias threads é suportado por qual classe e interface?

Classe Thread e pela interface Runnable.

3. Na criação de um objeto executável, por que pode ser melhor estender Thread em vez de implementar Runnable?

A principal vantagem é que ele pode se auto executar sem depender de chamadas externas, mas deve lembrar que o Java não possui heranças múltiplas, portanto, herdar da Thread pode ser uma limitação na sua implementação.

4. Mostre como podemos usar join() para esperar um objeto de thread chamado MyThrd terminar.

```
public class Main {  
    public static void main(String[] args) throws InterruptedException {  
        Thread t1 = new MyThrd();  
        t1.join();  
    }  
}
```

5. Mostre como configurar uma thread chamada MyThrd com três níveis acima da prioridade normal.

```
public class Main {  
    public static void main(String[] args) throws InterruptedException {  
        Thread t1 = new MyThrd();  
  
        t1.setPriority(Thread.NORM_PRIORITY + 3);  
  
        t1.join();  
    }  
}
```

6. Qual é o efeito da inclusão da palavra-chave synchronized em um método?

Faz com que esse método implemente o conceito de bloqueio, portanto, o Java bloqueia a execução deste quando a primeira thread chama esse método fazendo com que outra

Thread que chamar o mesmo método na mesma instância espere até a primeira execução terminar.