

## CEVAPLAR

### CEVAP 1:

Bir çok farklı kütüphaneye ait delete() methodu var. Fakat hiç biri bir objeyi destroy etmez. Finalize() methodu ise, GC çalıştığında bu method tetiklenir ve yapılacak son işlemler için bu method kullanılır. Objenin silinme işlemini Garbage collection yapar. **Cevap:F**

### CEVAP 2:

Program ilk çalıştığında try bloğu içerisinde aMethod() çağrılıyor. Bu method içerisindeki try bloğu içerisinde hata fırlatılıyor. Fakat bu hatayı yakalayacak catch metodu yazılmamış. Finally bloğu ise, catch metodu olsa da olmasa da çalışır. Dolayısıyla aMethod()'da, finally bloğundan dolayı ekrana "finally" yazar ve catch bloğu olmadığı için hatayı, kendisini kullanana method döner. Main method'da aMethod()'un fırlattığı hata, catch bloğunda handle edilir ve ekrana "exception" yazılır. Daha sonra try-catch bloklarından bağımsız olarak "finished" ekrana yazılmış. **Cevap: C**

### CEVAP 3:

Bu soruda Exception1, Exception0 'ı extends ediyor. Exception0 ise Exception'u extends ediyor. Dolayısıyla try bloğunda hata fırlatıldığında, ilk olarak Exception1'i handle eden catch bloğu varsa çalışır. Eğer handle edilmemişse, Exception0'ın handle edildiği catch bloğu çalışır. Eğer ikisi de handle edilmemişse Exception'ın handle edildiği blok çalışır. **Cevap:A**

### CEVAP 4:

Bri class, başka classlar tarafından extends edildiğinde ve methodları override edilirse, methodların modifierlerinin değiştirilmemesi gerekir. Eger public tanımlanmışsa, subclasslarda da public olmalı. Bu soruda getFloat() methodu abstract sınıfta public iken, AbstractTest methodunda private yapmış. **Cevap: D**

### CEVAP 5:

```
public double calculateSalary(int years, double base) {
    double salary=0;
    if(years<=5) {
        salary=base;
    }
    else if(years>5&&years<15){
        salary=base+base*(years-5)*10/100;
    }
    else {
        salary=base*2;
    }
    return salary;
}
```