Soru 3: Abstract class nedir? Interface nedir?

Abstract Class:

 Abstract class, doğrudan nesne oluşturulamayan ve diğer sınıfların kalıtım alarak kullanabileceği bir sınıftır.

 Normal metodlar ve soyut (abstract) metodlar içerebilir.

 Alt sınıflar, soyut metodları (abstract methods) override etmek zorundadır.

 Nesne üretilmez, yalnızca miras alınır (extends).

 abstract anahtar kelimesi ile tanımlanır.

Interface:

 Interface, tamamen soyut metodlardan oluşan bir yapıdır.

 Bir sınıf birden fazla interface'i implements ile uygulayabilir.

 Normal metod içeremez (Java 8’den sonra default metodlar eklenmiştir).

 interface anahtar kelimesi ile tanımlanır.

 Nesne üretilemez, sadece implement edilir.

Soru 4: İkisi arasındaki farklar.

|  |  |  |
| --- | --- | --- |
| **Özellik** | **Abstract Class** | **Interface** |
| **Soyut Metod İçerebilir mi?** | **Evet** | **Evet** |
| **Normal Metod İçerebilir mi?** | **Evet** | **Sadece default veya static metodlar (Java 8+)** |
| **Constructor İçerebilir mi?** | **Evet** | |  | | --- | |  |  |  | | --- | | **Hayır** | |
| **Çoklu Kalıtım (Multiple Inheritance)?** | **Hayır (extends ile tek sınıf miras alınabilir)** | **Evet (implements ile birden fazla interface uygulanabilir** |
| **Kullanım Şekli** | **extends ile miras alınır** | **implements ile uygulanır** |
| **Metodların erişim belirleyicisi** | **public, protected veya private olabilir** | **Tüm metodlar otomatik olarak public abstract olur** |
| **Değişkenlerin Erişim Türü** | **Normal değişkenler içerebilir** | **Tüm değişkenler public static final olur (yani sabittir)** |
| **Ne zaman kullanılır?** | |  | | --- | |  |  |  | | --- | | **Ortak özelliklere sahip nesneler için** | | **Birden fazla sınıfın aynı yetenekleri paylaşmasını sağlamak için** |