

# H4\_hw

1. C# dilinde kalıtım farklı türlerde uygulanabilmektedir. Ancak **\_Multiple inheritance** tipinde kalıtım desteği C# yoktur.
2. Bir sınıftan kalıtım ile devir alma işlemi için **:\_** operatörü kullanılmaktadır.
3. Bir üyenin alt sınıflarda geçersiz kılınabilmesini sağlamak üzere **\_ override** \_\_\_\_ anahtar kelimesi kullanılır.
4. Bir üyenin alt sınıflarda geçersiz kılınmasını engellemek üzere **\_sealed** anahtar kelimesi kullanılır.
5. Bir üyenin geçersiz kılınmasını sağlamak üzere **\_ virtual** \_\_\_\_ anahtar sözcüğü kullanılır.
6. static metotlar ve fields üyeleri **\_ virtual** \_\_\_\_ anahtar kelimesi ile deklare edilemezler.
7. Bir sınıftan devir alınarak tanımlanmış olan sınıf; bir başka sınıf tarafından devir alınıyor ise **\_ Singleinheritance** \_kalıtım türünün uygulandığı söylenebilir.
8. Temel sınıfta yer alan bir metot; devir alınan sınıfta gizlenmek isteniyor ise **\_ \_\_\_\_** anahtar kelimesi kullanılmalıdır.
9. Türetilmiş bir sınıfta temel sınıfta yer alan bir metot çağrılacak ise **new** \_anahtar kelimesi kullanılmalıdır.
10. Türetilmiş bir sınıftan temel sınıfın yapılandırıcı metoduna parametre geçilecek ise **base** anahtar kelimesi kullanılmalıdır.
11. **static** sınıflardan new anahtar sözcüğü ile nesne üretmek mümkün değildir.
12. Gövdesi olmayan ve bir sınıf içerisine yazılan metotlar **\_ \_** metot olmalıdır.
13. Temel sınıfta **\_ \_**olarak tanımlanmış olan bir metot; mutlaka devir alınan sınıfta geçersiz kılınmalıdır.

14. Bir sınıfın kalıtım ile devir alınmasını engellemek için \_ **sealed**\_\_\_\_ anahtar kelimesi kullanılır.

15. Sınıf içinde yapılandırıcı metotlar arasında parametre geçişi sağlanacak ise \_ **base** anahtar sözcüğü kullanılmalıdır.