**HOMEWORK ASSIGNMENT 3.1**

**SORU 1:**

What is the difference between Step Over and Step Into?

**Cevap:**

**Step Over:** Metod çağrısına girmeden bir sonraki satıra geçmek için kullanılır. Örneğin, debug moddayken sırada çalıştırılması gereken bir metod vardır. Ancak biz bu metodun içine girmeden bir sonraki satıra geçmesini istiyorsak, debug modda step over kullanılmalıdır.

**Step Into:** Metod çağrısındaki, metodun içine girerek ilk satırına ilerlemek için kullanılır. Örneğin, debug moddayken sırada çalıştırılması gereken bir metod vardır. Bu metodun içine gidip ilk satırından itibaren nasıl çalıştığını görmek istiyorsak, debug modda step into kullanılmalıdır.

Kısaca, metodların içine girmek için step into, atlamak için ise step over kullanılır.

**SORU 2:**

How does each part of the debugger view change when you step into a new method?

**Cevap:**

Debugger view, stack trace i gösterir. Stack trace içeriği ise içine girilen metodun değişkenlerini göstererek güncellenir. Ayrıca metodun bulunduğu sınıf açılır ve vurgulanır. Yani içine girilen metodun yeri ve sınıfı developer a fark ettirilir.

**HOMEWORK ASSIGNMENT 3.2**

No question.

**HOMEWORK ASSIGNMENT 3.3**

**SORU 1:**

Based on your exploration in the debugger, how does the Android platform decide which implementation to use in the Compat classes?

**Cevap:**

Android hangi implementasyona, android sürümünün yapı numarasına göre (Build number of android version) karar verir. Bu da ContextCompact sınıfında belirtilir.

**SORU 2:**

Besides the differences in the method signatures, what is the difference between the implementations? Why is a compatibility class required at all?

**Cevap:**

Uyumluluk sınıfları (compatibility classes), uygulamaların farklı android sürümlerine sahip cihazlarda amaçlandığı gibi çalışacağından emin olmak için gereklidir. Bu sınıflar yeni özelliklerin eski olanlarla entegre edilmesini sağlar.