

سوال اول (

۱) نادرست . primitive تایپ ها برای ذخیره شدن در کالکشن هایی که عموماً جنریک تایپ نیز هستند نیاز به باکسینگ و تبدیل به آبجکت معادل آن پرمیتیو تایپ دارند و پس از این تبدیل به صورت آبجکت ذخیره میشوند .

۲) درست

استرینگ در جاوا به دلایل زیر immutable است :

امنیت : ثابت ماندن اطلاعات در دیتابیس ( مانند یوزر نیم , پسورد و ... )  
سرعت و کش : اگر در کامپایلر موارد زیر نوشته شود تنها یک آبجکت استرینگ test ساخته میشود .

```
a = "test" ;  
b = "test" ;
```

۳) نادرست

در توضیحات set در جاوا آمده است :

Adds the specified element to this set if it is not already present [...] If this set already contains the element, the call leaves the set unchanged .

بنا بر این زمانی یک المان در set قرار میگیرد که آن المان از قبل موجود نباشد و این اتفاق از duplicate جلوگیری میکند .

۴) نادرست

چون ساختار map مانند تابع ۱-۱ است و هر key به یک value نظیر میشود باید key ها یونیک باشند . زمانی که value جدیدی به یک key نسبت داده میشود , آن value جای مقدار قبلی را می گیرد .

سوال دوم (

Iterator :

ایتریتور برای itarate کردن و طی کردن یک collection در جاوا کاربرد دارد . با استفاده از ایتریتور میتوان به اعضای یک کالکشن و به صورت پشت سر هم دسترسی داشت .

Anonymous object :

زمانی که یک آبجکت از کلاسی ساخته میشود ولی به یک متغیر assign نمیشود به این آبجکت یک Anonymous object میگوییم . Anonymous آبجکت ها یکبار ساخته میشوند و فقط در همان لحظه وجود دارند و در کارهایی مانند استفاده از تابع آنها کاربرد دارند .

```
new myClass().show() ;
```

Class variables:

این نوع متغیرها که با static member variable نیز شناخته میشوند به صورت یک متغیر استاتیک در کلاس تعریف میشوند و از آنها تنها یک کپی ساخته میشود و بین تمامی instance های ساخته شده از آن کلاس share میشود و زمانی که مقدار این متغیر در یک آبجکت از آن کلاس تغییر کرد مقدار آن در کل آبجکت های آن کلاس تغییر میکند .

Heap :

بخشی از حافظه است که برای ذخیره سازی آبجکت ها و متغیر ها و موارد مربوط به آن آبجکت آن آبجکت استفاده میشود . وقتی که با استفاده از کلمه کلیدی new و یک constructor, یک آبجکت ایجاد می کنیم قسمتی از حافظه heap به این کار اختصاص داده میشود و در آن ذخیره میگردد .

Stack :

بخشی از حافظه است که برای ذخیره سازی تابع مانند متغیر ها , پارامتر ها و موارد مرتبط استفاده میشود . زمانی که یک متد invoke یا فراخوانی میشود بخشی از حافظه stack به این کار اختصاص می یابد . با فراخوانی تابع جدید این تابع بر روی تابع قبلی قرار میگیرد و زمانی که فراخوانی آن تمام شد از حافظه stack خارج میشود .

Unit testing :

یونیت تستینگ یک نوع تست برنامه است که در آن قسمت های مختلف و تشکیل دهنده سورس کد به صورت جدا جدا (واحد) چک میشوند .

سوال سوم (

کالکشن یک مجموعه از اینترفیس ها و کلاس ها است که برای ذخیره سازی و پردازش داده ها و مجموعه آبجکت ها استفاده میشود . اینترفیس کالکشن این امکان را میدهد که بتوان با گرهی از آبجکت ها کار کرد و به عنوان سر دسته در کالکشن ها قرار دارد . اینترفیس collection خود اینترفیس Iterable را به ارث برده است . اینترفیس های List , queue , Set , اینترفیس extend را collection extend میکند . کلاس های HashSet , LinkedHashSet , اینترفیس Set را implement میکنند . اینترفیس SortedSet نیز اینترفیس Set را extend میکند و سپس Tree ست آن را implement میکند . کلاس های ArrayList , Vector اینترفیس List را implement میکنند و کلاس PriorityQueue اینترفیس Queue را implement میکند . همچنین کلاس LinkedList دو اینترفیس List و Queue را implement میکند . پکیج java.util شامل کلاس ها و اینترفیس های فریمورک Collection است . اینترفیس Iterator امکان iterate کردن المان ها به سمت جلو را مهیا میکند که کالکشن نیز از آنها استفاده میکند .