

به نام خدا

سوالات اشکال زدایی

۹۷۳۹۰۲۱

محمد رضا حسن زاده

دستور کار کارگاه:

سوال اول

اشکال قطعه کد اول :

برای نوشتن تابع یا متد داخل برنامه باید به صورت `static <returnType> my_method()` نوشته شود. در صورتی که قطعه کد اول تابع را به صورت `public void my_method()` نوشته است. پس باید اصلاح شود به `static void my_method()`.

اشکال قطعه کد دوم:

اسم کلاس باید با حروف بزرگ شروع شود و همچنین باید با نام فایل یکسان باشد یعنی باید به صورت `public class Main` نوشته شود. همچنین باید کد ما یک `main` داشته باشد تا از آنجا شروع شود در نتیجه نیاز به `public static void main(String[] args)` نیز می باشد. همچنین برای استفاده از تابع `myanothermethod` در `main` برنامه باید به صورت `static` تعریف شود و همچنین اسم آن باید از قانون `camelCase` نیز پیروی کند و به صورت `static void myAnotherMethod()` باشد.

سوال دوم

الف) اگر سمی کالن قرار ندهیم به این معناست که ته کد هنوز بسته نشده و این یک خطای نحوی است (syntax error) در نظر گرفته می شود. و ارور آن به صورت `java: ';' expected` می باشد

ب) اگر دابل کوتیشن را نبندیم هم باعث میشود که یک رشته ی بسته نشده داشته باشیم. و ارور آن به صورت `java: unclosed string literal` می باشد

ج) اگر یکی از آکلاد ها را نبندیم این بدان معناست که اسکویی که مثلا برای `main` در نظر گرفتیم را هنوز نبستیم و این یک خطا است و ارور آن به صورت `java: reached end of file while parsing` می باشد.

سوال سوم

تابع `main` در ورودی های خود یک `String[] args` دریافت می کند و اگر که این عبارت را داخل پرانتز `main` ننویسیم کامپایلر به ما ارور می دهد که تابع `main` به صورت `public static void main(String[] args)` باید تعریف شود در نتیجه مشکل از ورودی های تابع `main` می باشد.

سوال چهارم

جاوا در انتهای توسعه ی نرم افزار آن کامپایل می شود و تبدیل به زبان `java bytecode` ولی در زمان اجرای برنامه ای که به زبان جاوا نوشته شده است همانند زبان های مفسری می شود در نتیجه جاوا هم یک زبان مفسری است و هم یک زبان کامپایلری