



برنامه نویسی پیشرفته

زمستان و بهار ۹۹-۱۳۹۸ - دانشکده علوم ریاضی

دانشگاه صنعتی شریف

با توجه به شرایط خاص پیش آمده تیم درس برنامه نویسی پیشرفته تصمیم گرفتند که یک پرسشنامه بدون تاثیر در ارزیابی برای بررسی میزان پیشرفت مطالعه برگزار کند. هدف از این پرسشنامه بررسی پیشرفت عملکرد و مطالعه شما است. هدف از این پرسشنامه این موارد است:

- جبران فاصله ایجاد شده میان دانشجویها با همدیگر که امکان ارائه بازخورد پیشرفت مطالعه به یکدیگر را ایجاد می‌کند.
- جبران فاصله ایجاد شده میان دانشجویها و تیم درس برای دریافت بازخورد پیشرفت تحصیلی
- بازخورد هر دانشجو به خود در رابطه با پیشرفت مناسب در مطالعه و یادگیری مفاهیم از طریق منابع درس
- جهت‌دهی به اشکالاتی که شما ممکن هست هنوز در جریان وجود نقطه ضعف خود در این رابطه نباشید و رفع آنها در جلسه‌های رفع اشکال آنلاین
- دقت کنید که این پرسشنامه تنها مرجع برای بازخورد پیشرفت مناسب شما در مطالعه منابع نیست. تمرین‌ها و پروژه‌ها نیز سهم بزرگی در این مساله دارند. پس حتما پس از پاسخ به سوال‌ها و ارسال پاسخ‌ها، اشکال‌ها و ابهام‌هایی که داشتید در جلسه‌های آنلاین رفع اشکال در میان بگذارید و رفع کنید و از این فرصت استفاده کنید.

توضیحات

- نتیجه این پرسشنامه تاثیری در ارزیابی نهایی این درس ندارد.
- این پرسشنامه برای اطمینان بیشتر از اینکه مسیر درس را درست طی می‌کنید طراحی شده.
- اگر نیاز به بررسی صحت پیشرفتتان دارید حتما در این پرسشنامه شرکت کنید.
- در صورتی که با مطالب درس به درستی پیش آمده باشید می‌توانید به تمام سوال‌ها پاسخ دهید.
- در صورتی که به بخشی از هر سوال تسلط ندارید یا احتیاج به بررسی صحت پاسخ‌ها دارید حتما در جلسه‌های رفع اشکال شرکت کنید و اشکال یا ابهام‌های خود را رفع کنید.
- سعی کنید جواب‌ها کوتاه و دقیق باشند که مرور جواب در جلسه رفع اشکال سریع‌تر انجام شود.
- از آنجایی که این پرسشنامه برای یادگیری طراحی شده می‌توانید در پر کردن سوال‌ها با هر فردی مشورت و همفکری کنید.

نحوه انجام پرسشنامه

- برای پاسخ به این پرسشنامه یک نسخه از این فایل را از منو فایل و گزینه گرفتن یک کپی برای خود ایجاد کنید و جواب‌های آن را در همین فایل بنویسید.
- پس از جواب دادن به سوال‌ها آن را در قالب PDF دانلود کنید.
- فایل PDF در یک ریپازیتوری [github](#) بارگذاری کنید.
- آدرس این ریپازیتوری را در یک فایل یک خطی با پسوند جاوا داخل کوئرا و در بخش پرسشنامه بررسی پیشرفت بارگذاری کنید.

سوال‌ها

سوال ۱

خروجی این برنامه را بدست بیاورید و به ازای هر خط توضیح دهید که چرا به این خروجی رسید؟

```
class Classes {
    static class A {
        static int intValue = 0;
        int integerValue = 20;

        A() {
            integerValue = 5;
            printValue();
            print();
        }

        void printCaller() {
            print();
        }

        void printValue() {
            System.out.println("B:" + integerValue);
        }

        void print() {
            System.out.println("A:" + intValue);
        }
    }

    static class B extends A {
        B(int v) {
            intValue = v;
            integerValue = 15;
            printValue();
            print();
        }

        void print() {
            System.out.println("B:" + intValue);
        }

        void printSuper() {
            super.print();
        }

        void printCaller() {
            printValue();
            super.printValue();
        }
    }
}
```

```

    void printValue() {
        System.out.println("B:" + integerValue);
        super.printValue();
    }
}

static public class C extends A {
    void printCaller() {
        System.out.println("B:" + integerValue);
    }

    void print() {
        System.out.println("A:" + intValue);
        super.printCaller();
    }
}
}

class Problem1 {
    public static void incrementValue(Classes.A object) {
        object.intValue++;
        object.integerValue++;
    }

    public static void incrementValue(int firstValue, int secondValue) {
        firstValue++;
        secondValue++;
    }

    public static void main(String[] args) {
        Classes.A a = new Classes.A();
        وقتی از کلاس A شی ساخته شود با توجه به constructor کلاس مقدار integerValue و intValue که برابر ۵ و ۰ هستند پرینت می شوند (دو خط اول خروجی).
        Classes.B b = new Classes.B(10);
        با توجه به این که کلاس A، کلاس پدر است و زمانی که شی ساخته می شود، ابتدا constructor کلاس A صدا می شود. اول تابع printValue اعمال می شود که منجر به خروجی سوم و چهارم شده (در این تابع، تابع printValue کلاس پدر نیز فراخوانی می شود) و سپس تابع print فراخوانی شده که منجر به خروجی پنجم می شود. سپس constructor کلاس B صدا زده می شود که منجر به خروجی های ۶ و ۷ و ۸ می شود.
        Classes.A c = b;

        b.print();
        تابع print روی شی b از کلاس B فراخوانی می شود که منجر به خروجی ۹ ام می شود.
        c.print();
        شی c در اصل از کلاس B آمده است پس با فراخوانی متد print روی آن در واقع متد override شده در کلاس فراخوانی می شود (خروجی ۱۰ ام)
        ((Classes.A) b).print();
        مانند خط بالا

        b.printSuper();
        تابع print در کلاس پدر را فراخوانی می کند (خروجی ۱۲ ام)

        a.printCaller();
        مانند خط بالا تابع print کلاس پدر فراخوانی می شود
    }
}

```

```

b.printCaller();

```

خروجی ۱۴ و ۱۵

ابتدا تابع `printValue` در کلاس B و سپس همین تابع در کلاس پدر فراخوانی می‌شود

```

c.printCaller();

```

مانند خط بالا

```

incrementValue(a);
a.printCaller();

```

در خط بالا با توجه به فراخوانی تابع `incrementValue` مقدار `intValue` (با توجه به استاتیک بودنش) یک واحد اضافه می‌شود و این خط منجر به خروجی ۱۸ ام می‌شود

```

incrementValue(b);
b.printCaller();

```

مقدار `integerValue` یک واحد اضافه شده و تبدیل به ۱۶ می‌شود. این خط منجر به خروجی ۱۹ و ۲۰ و ۲۱ ام می‌شود.

```

incrementValue(c);
c.printCaller();

```

مثل خط قبل با این تفاوت که `integerValue` تبدیل به ۱۷ شده است.

```

incrementValue(b.intValue, b.integerValue);

```

این تابع در پایان هیچ تاثیری در مقدار نهایی ورودی هایش نمی‌گذارد و آن‌ها هیچ تغییری نمی‌کنند پس `integerValue` همان ۱۷ می‌ماند

```

b.printCaller();
c.printCaller();

```

منجر به ۶ خط آخر خروجی می‌شوند.

```

}
}

```

خروجی:

```

B:5
A:0
B:5
B:5
B:0
B:15
B:15
B:10
B:10
B:10
B:10
A:10
A:10
B:15
B:15
B:15
B:15
B:15
A:11
B:16
B:16
B:16
B:17
B:17
B:17

```

B:17
B:17
B:17
B:17
B:17
B:17

سوال ۲

توضیح دهید که هدف از ارث بری در شی گرای چيست. چه زمان از composition و چه زمان از inheritance استفاده می‌کنیم؟ چگونه می‌توانیم از سازنده پدر را فراخوانی کنیم؟ چگونه می‌توانیم سازنده دیگری از خود کلاس را فراخوانی کنیم؟ -از کلاس پدر با کلمه کلیدی super سازنده را فراخوانی می‌کنیم و از خود کلاس با استفاده از کلمه کلیدی this. البته برای فراخوانی توابعی که در هر دو کلاس وجود دارند اگر از هیچ کلمه کلیدی استفاده نکنیم به صورت خودکار تابع خود کلاس فراخوانی می‌شود.

زمانی که تمام اشیا یک کلاس تمام ویژگی‌های یک کلاس دیگر را داشته باشند به علاوه ویژگی‌های مخصوص به خود، از inheritance استفاده می‌کنیم و اگر کلاس ما بخواهد ویژگی‌ای از جنس یک کلاس دیگر داشته باشد از composition استفاده می‌کنیم.

سوال ۳

توضیح دهید که چرا از رابط‌ها (interface) استفاده می‌کنیم. چه محدودیت‌هایی نسبت به یک کلاس دارند و چرا امکان پیاده‌سازی متد در آنها داده شده است؟

در آن‌ها عموماً متغیری تعریف نمی‌شود، از آن‌ها شی ساخته نمی‌شود، کلیات رفتار شی را بررسی می‌کند و وارد نحوه پیاده‌سازی متد ها نمی‌شود.

سوال ۴

کلاس انتزاعی (abstract) چیست و چه زمانی در مدل‌سازی از یک کلاس انتزاعی استفاده می‌کنیم؟ این نوع کلاس چه تفاوتی با رابط (interface) دارد؟

-کلاس انتزاعی کلاسی است که ما امکان ساختن شی‌ای از این کلاس را نداریم. وقتی از این کلاس استفاده می‌کنیم که تابعی انتزاعی داشته باشد یا نخواهیم از این کلاس شی‌ای ساخته شود.

یک interface می‌تواند از چندین interface دیگر ارث بری کند اما یک کلاس نمی‌تواند. در یک کلاس انتزاعی filed ها و ویژگی‌هایی نیز تعریف می‌شود اما کار interface تولید محتوا نیست و صرفاً یک واسط است.

سوال ۵

override کردن تابع و متغیر چه تاثیری در عملکرد متد در یک کلاس فرزند می‌گذارد؟ چطور می‌توانیم پس از override شدن یک متد در کلاس فرزند در هر کدام از مکان‌های زیر به نسخه هم نام آن متد در کلاس پدر دسترسی پیدا کنیم؟

- متدی داخل کلاس پدر: با فراخوانی آن متد، متد با همان تعریف کلاس پدر فراخوانی می‌شود.
- متدی داخل کلاس فرزند: با استفاده از کلمه super
- خارج از دو کلاس: با فراخوانی کردن متد روی شی‌ای از جنس کلاس پدر.

--سوال ۶

توضیح دهید که منظور از چندریختی در شی گرایی چیست و چه مزیتی ایجاد می‌کند.
ممکن است تمام اشیا کلاس پدر از یک ویژگی برخوردار باشند اما این ویژگی در هر یک منحصر به فرد باشد. به عنوان مثال تمام حیوان ها از ویژگی حرکت کردن برخوردار هستند اما هر نوع حیوانی حرکت متفاوتی دارد. به این ویژگی در شی گرایی چندریختی گفته می‌شود.

سوال ۷

چرا از توابع و متدها در زبان برنامه نویسی استفاده می‌کنیم؟ در طراحی برنامه و شکستن آن به توابع و متدهای مختلف چه نکته‌هایی را باید رعایت کرد که خوانایی آن بیشتر شود و پیچیدگی اضافی نداشته باشیم؟

سوال ۸

کلاس درونی (inner class) چه انواعی دارد و هر کدام چه کاربردی در مدل‌سازی و توصیف موجودات دارد؟ چگونه می‌توانیم یک شی از هر نوع ایجاد کنیم؟ در صورت override شدن یک متد یا متغیر توسط یک کلاس درونی چگونه می‌توان به نسخه override شده از کلاس بیرونی دسترسی پیدا کرد؟

--سوال ۹

- کلمه کلیدی final روی هر کدام از موارد زیر چه تاثیری دارد؟
- تابع و متد : دیگر قابل override شدن نمی‌باشند.
 - تعریف کلاس: دیگر نمی‌توانند زیر کلاسی داشته باشند.
 - یک متغیر از نوع شی: فیلد ها و محتوایش قابل تغییر نیستند.
 - یک متغیر از نوع پایه: مقدار آن قابل تغییر نیست.

سوال ۱۰

- کلمه کلیدی static روی هر کدام از موارد زیر چه تاثیری دارد؟
- تابع و متد
 - تعریف کلاس
 - یک متغیر از نوع شی
 - یک متغیر از نوع پایه