



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();
        Classes.B b = new Classes.B(10);
        Classes.A c = b;

        b.print();
        c.print();
        ((Classes.A) b).print();
        b.printSuper();
        a.printCaller();
        b.printCaller();
        c.printCaller();
        incrementValue(a);
        a.printCaller();
        incrementValue(b);
        b.printCaller();
        incrementValue(c);
        c.printCaller();
        incrementValue(b.intValue, b.integerValue);
        b.printCaller();
        c.printCaller();
    }
}

```

سوال ۲

توضیح دهید که هدف از ارث بری در شی گرای چ چیست. 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؟

پاسخ: هدف از ارث بری استفاده دوباره از کدهای کلاس پدر در کلاس فرزند است. هنگامی که یک آبجکت با یک آبجکت دیگر رابطه زیرمجموعه ای دارد از inheritance و هنگامی که یک آبجکت از یک آبجکت دیگر تشکیل شده باشد از composition استفاده می‌کنیم. با استفاده از super پدر را فراخوانی می‌کنیم. با استفاده از اسم خود کلاس سازنده دیگری را صدا 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؟

پاسخ: برای بدست تعریف کردن مفاهیم و داشتن چند inheritance هم‌زمان. نمیتوان در آن‌ها constructor داشت. برای کار کردن کد های قدیمی تر و خراب نشدن کد ها با اضافه کردن یک متد به interface.

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؟

پاسخ: کلاسی است که نمیتوان آن را ساخت و constructor ندارد. هنگامی که مدل ما به صورت مستقل وجود نداشته باشد. نوع پیاده سازی آن و قابلیت اکستند کردن کلاس و پیاده سازی interface و نوع فیلد های آن ها.

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
- متدی داخل کلاس فرزند
- خارج از دو کلاس

پاسخ: با override کردن متد فرزند کاری را میکند که در تابع override شده ذکر شده است. یک با صدا کردن اسم تابع (دو) با صدا کردن super.function_name سه با صدا کردن اسم تابع

سوال ۶

توضیح دهید که منظور از چندریختی در شی گرای چیست و چه مزیتی ایجاد می‌کند.

پاسخ: چندریختی حالتی است که برای یک آبجکت چند حالت دارد مثلاً یک شکل هم میتواند دایره باشد هم مثلث هم مربع هم مزیت آن پویا شدن و افزایش انعطاف آن میباشد.

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؟

پاسخ: تا جلوی زدن کد تکراری گرفته شود و خوانایی برنامه افزایش یابد. نام گذاری درست و کوتاه نگه داشتن تابع ها.

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؟

پاسخ: نوع 1) کلاس داخلی ساده: این نوع کلاس مانند کلاس معمولی داخل یک کلاس تعریف می‌شود. نوع 2) کلاس داخلی استاتیک: این کلاس داخل یک کلاس دیگر با پیشوند استاتیک ساخته می‌شود. نوع 3) کلاس داخلی متد: این کلاس داخل یک متد در داخل یک کلاس دیگر به تعریف می‌شود. نوع 4) کلاس داخلی ناشناس: این کلاس داخل یک کلاس دیگر به دو صورت تعریف می‌شود: 1) پیاده سازی یک اینترفیس 2) subclass یک کلاس دیگر. در صورت override شدن می‌توانیم با super دسترسی داشته باشیم.

سوال ۹

کلمه کلیدی final روی هر کدام از موارد زیر چه تاثیری دارد؟

- تابع و متد
- تعریف کلاس
- یک متغیر از نوع شی
- یک متغیر از نوع پایه

پاسخ: 1) جلوگیری از override شدن 2) جلوگیری از extends شدن 3) جلوگیری از دوباره شروع شدن ولی جلوی تغییر فیلدهایش را نمی‌گیرد. 4) جلوگیری از تغییر دادن مقدار اولیه

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

پاسخ: 1) آن تابع را می‌توان به صورت جدا و قبل از ساختن کلاس استفاده کرد. 2) فقط روی کلاس های داخلی می‌تواند تعریف می‌شود و باعث می‌شود تا فقط بتواند به متغیرهای استاتیک کلاس بالاترش دسترسی داشته باشد. 3 و 4) از آن متغیر دقیقاً یکی ساخته می‌شود و در همه جا صدا زده می‌شود.