



# 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();
    }
}

```

B:5 چون وارد کانستراکتور کلاس A می شود و از انجا تابع pirintvalue صدا زده می شود

A:0 چون وارد کانستراکتور کلاس A می شود و از انجا تابع pirint صدا زده می شود

B:5 چون از B وارد کانستراکتور کلاس A می شود و از انجا تابع pirintvalue صدا زده می شود

B:5 به خاطر خط `super.printValue()` است

B:0 تابع print در کلاس B

B:15 کانستراکتور کلاس B

B:15 کانستراکتور کلاس B

B:10 تابع print در کلاس B

B:10 صدا زدن `b.print();`

B:10 صدا زدن `c.print();`

B:10 صدا زدن `b.print())Classes.A)`

A:10 تابع print در کلاس A

A:10 صدا زدن `a.printCaller();`

B:15 صدا زدن `b.printCaller();`

B:15 صدا زدن `b.printCaller();`

B:15 صدا زدن `b.printCaller();`

B:15 صدازدن c.printCaller();

B:15 صدازدن c.printCaller();

B:15 صدازدن c.printCaller();

A:11 صدازدن a.printCaller(); که در خط بالا مقدار A تغییر می کند

B:16 ضدازدن b.printCaller(); که مقدار آن در خط بالا تغییر می کند

B:16 به خاطر b.printCaller();

B:16 به خاطر b.printCaller();

B:17 ضدازدن c.printCaller(); که مقدار آن در خط بالا تغییر می کند

B:17 ضدازدن c.printCaller();

B:17 ضدازدن c.printCaller();

B:17 به خاطر b.printCaller();

B:17 به خاطر b.printCaller();

B:17 به خاطر b.printCaller();

B:17 ضدازدن c.printCaller();

B:17 ضدازدن c.printCaller();

B:17 ضدازدن c.printCaller();

## 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؟

هدف ارث بری این است که از تعریف تابع ها و ویژگی های تکراری برای اشیا صرف نظر کنیم و ویژگی ها و رفتار هایی که پدران آنها دارند هم دارند و نیازی به تکرار آنها نیست. وقتی که می‌خواهیم از یک شیء، بعنوان یک پروپرتی در یک شیء دیگر استفاده کنیم composition و زمانی که بخواهیم یک شیء تمام ویژگی های شیء دیگر را داشته باشد از inheritance با استفاده از کلید واژه super می‌توان به کلاس و کانستراکتور کلاس پدر دسترسی داشت اگر چند نوع سازنده داشت باشد می‌توان هر کدام را با توجه به ورودی که به سازنده می‌دهی فراخوانی کنیم.

## 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؟

برای دسترسی به ویژگی های هر شی از رابط‌ها استفاده می‌کنیم و این بخشی از کلاس است که رفتار های اون شی در آن قرار می‌دهیم و برای دسترسی کاربران به ویژگی های هر شی هست

## 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؟ زمانی که می‌دانیم چه کارهایی باید انجام شود اما نمی‌دانیم چطوری باید انجام شود از کلاس های انتزاعی استفاده می‌کنیم و در این کلاس ها می‌دانیم چه رفتار هایی دارد ولی چگونه رفتار می‌کنند معلوم نیست پس صرفاً در ن تعریف تابع می‌آید نه پیاده سازی آن

## 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

## سوال ۶

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

## 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؟  
برای راحت تر کردن کار و تمیزتر شدن کد و باید اسم ها خوانا باشد و با توجه به کارکرد آنها خروجی برگرداند یا نه

## 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 بدنه یا پیاده سازی آن هیچگاه قابل تغییر نمی باشد
- تعریف کلاس قابلیت وراثت را از دست می دهد
- یک متغیر از نوع شی به متغیر دیگری نمی واند ارجاع داده شود
- یک متغیر از نوع پایه مقدار ان قابل تغییر نیست

## سوال ۱۰

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

- تابع و متد به اون شی مربوط نیست و به کلاس مربوط است و بدون ساختن شی از اون کلاس می توان انرا فراخوانی کرد
- تعریف کلاس یک کلاس static نمیتوان نمونه ساخت
- یک متغیر از نوع شی به اون شی مربوط نیست و به کلاس مربوط است
- یک متغیر از نوع پایه به اون کلاس مربوط نیس و