

# پروژه پایانی درس مبانی برنامه‌نویسی

## فهرست مطالب

1	مقدمه
1	تعریف پروژه
1	• روند برنامه
2	• نمایش برنامه هفتگی
2	• پیاده‌سازی
2	• ذخیره سازی
2	• مثال
6	• نکات پیاده سازی
6	• بخش‌های امتیازی

## مقدمه

در این پروژه قصد داریم یک برنامه‌ریز هفتگی را پیاده‌سازی کنیم. این برنامه‌ریز یک هفته را در اختیار کاربر قرار می‌دهد و او می‌تواند کارهای هفتگی خود را در آن وارد کند.

## تعریف پروژه

### • روند برنامه

کاربر به برنامه وارد می‌شود. در ابتدا وی می‌بایست نام خود را وارد کند و پیام خوش آمدگویی را می‌بیند. وی می‌تواند برنامه‌ی هفتگی خود را ببیند و به آن برنامه‌های جدید اضافه کند. برنامه به کاربر اجازه می‌دهد که در صورت نیاز برنامه‌های قبلی خود را پاک کند. هر برنامه کاربر از اجزای زیر تشکیل شده است:

- ❖ نام (نام هر برنامه برای یک روز یکتا می‌باشد. به طور مثال کاربر برای روز شنبه تنها می‌تواند یک برنامه ورزش داشته باشد).
- ❖ زمان شروع که به صورت عدد صحیح بوده و زمان شروع را به دقیقه نسبت به ساعت ۱۲ شب اعلام می‌کند. (به طور مثال اگر برنامه ورزش کاربر از ساعت ۸ آغاز شود زمان شروع آن برابر با  $8 * 60$  می‌باشد).
- ❖ زمان پایان که به صورت عدد صحیح بوده و زمان پایان را به دقیقه نسبت به ساعت ۱۲ شب اعلام می‌کند.

کاربر برای پاک کردن یک برنامه از نام و روز آن استفاده می‌کند و برای ساخت یک برنامه جدید پارامترهای فوق را به همراه روز آن را وارد می‌کند. روز یک عدد صحیح می‌باشد که صفر برابر با شنبه، یک برابر با یکشنبه و ... است.

## • نمایش برنامه هفتگی

فرض کنید کاربر یک برنامه با نام Exercise برای روز شنبه از ساعت ۸ تا ۱۰ و برنامه دیگری با نام Study برای ساعت ۱۰ تا ۱۲ همان روز دارد، برنامه هفتگی وی به شکل زیر نمایش داده می‌شود:

```
0 (Saturday): Exercise (from: 480, to: 600) -> Study (from: 600, to: 720)
1 (Sunday) :
2 (Monday) :
3 (Tuesday) :
4 (Wednesday) :
5 (Thursday) :
6 (Friday) :
```

## • پیاده‌سازی

- کاربر می‌تواند به تعداد دلخواه برنامه برای هر روز داشته باشد.
- در صورتی که برنامه‌ای با نام تکراری برای یک روز وارد شوند برنامه می‌بایست با پیام مناسب عملیات درج را انجام ندهد. (نام برنامه‌ها در طی یک روز یکتا می‌باشند).
- کاربر نمی‌تواند برنامه‌هایی با تداخل زمانی برای یک روز داشته باشد و در صورتی که این اتفاق رخ دهد برنامه می‌بایست با پیام مناسب عملیات درج را انجام ندهد.

## • ذخیره سازی

در صورت نیاز کاربر باید بتواند برنامه‌ی هفتگی خود را ذخیره کرده و تا با اجرای دوباره برنامه بتواند آن را ببیند. برای این امر برنامه هر کاربر به اسم او ذخیره می‌شود و در آینده می‌تواند در برنامه آن را دوباره بارگذاری کند. هر کاربر حداکثر می‌تواند یک برنامه را ذخیره کند و برنامه‌های بعدی در صورت ذخیره‌سازی، بر روی قبلی‌ها نوشته می‌شوند.

## • مثال

برای مثال روند کلی برنامه یکبار مرور می‌کنیم:

```
Enter your name:
```

```
> Parham
```

```
Welcome Parham!
```

```
[1] new task
```

```
[2] show tasks
```

```
[3] save
```

```
[4] load
```

```
[5] quit
```

```
> 1
```

```
Enter the day:
```

```
> 0
```

```
Enter the name:
```

```
> Exercise
```

```
Enter the start time:
```

```
> 600
```

```
Enter the end time:
```

```
> 720
```

```
There is no error and the new task is created
```

```
[1] new task
```

```
[2] show tasks
```

```
[3] save
```

```
[4] load
```

```
[5] quit
```

```
> 2
```

```
0 (Saturday): Exercise (from: 480, to: 600)
```

```
1 (Sunday):
```

2 (Monday) :

3 (Tuesday) :

4 (Wednesday) :

5 (Thursday) :

6 (Friday) :

[1] new task

[2] show tasks

[3] save

[4] load

[5] quit

> 1

Enter the day:

> 0

Enter the name:

> Exercise

Enter the start time:

> 720

Enter the end time:

> 840

Conflict with name

[1] new task

[2] show tasks

[3] save

[4] load

[5] quit

> 3

```
[1] new task
[2] show tasks
[3] save
[4] load
[5] quit
> 5
```

Bye Parham

Enter your name:

> Parham

Welcome Parham!

```
[1] new task
[2] show tasks
[3] save
[4] load
[5] quit
> 4
```

```
[1] new task
[2] show tasks
[3] save
[4] load
[5] quit
> 2
```

0 (Saturday): Exercise (from: 480, to: 600)

1 (Sunday):

2 (Monday):

3 (Tuesday) :

4 (Wednesday) :

5 (Thursday) :

6 (Friday) :

## • نکات پیاده سازی

۱. تا جایی که می‌توانید پروژه خود را به توابع کوچکتر بشکنید.
۲. به هیچ وجه از goto استفاده نکنید!
۳. توصیه می‌شود پیاده‌سازی خود را به صورت مرحله به مرحله تست کنید و تست آن را به آخر موکول نکنید.
۴. مانند همه‌ی تمرین‌ها، پروژه را نیز در کوئرا آپلود می‌کنید و تقلب‌ها با نرم افزار MOSS گرفته شده و تقسیم نمره صورت می‌گیرد.
۵. برنامه‌ی شما روی سیستم خودتان تست می‌شود بنابراین می‌توانید از هر کتابخانه‌ای استفاده کنید.

## • بخش‌های امتیازی

۱. نمایش تسک‌ها به ترتیب زمان آغاز آن‌ها
۲. نمایش صحیح خطاهای سیستم (مثلا باز نشدن فایل، پیدا نشدن فایل و ...)
۳. [Github](#) و [Gitlab](#) سایت‌هایی مبتنی بر کنترل کننده‌ی نسخه‌ی Git می‌باشد. شما به وسیله‌ی Git می‌توانید در هر مرحله کد خود و تغییرات در آن را به صورت مستند ذخیره کنید و نیازی نیست به صورت دستی از کد خود کپی بگیرید. در صورتی که به طرز صحیح از این سایت استفاده کنید نمره‌ی امتیازی به شما تعلق می‌گیرد. فقط توجه داشته باشید از مخزن‌های private استفاده کنید.
۴. هر ایده‌ی دیگری که احساس می‌کنید باعث جذاب‌تر شدن بازی می‌شود.

موفق باشید. ☺