



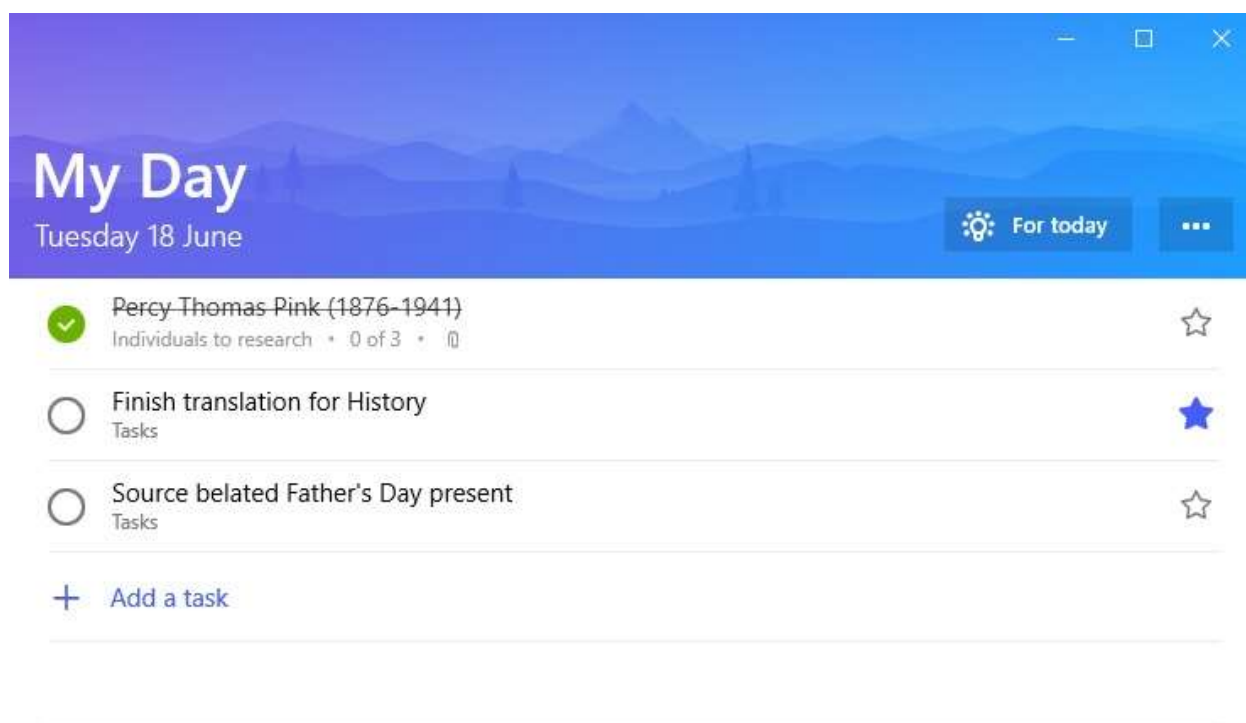
تمرین پنجم \_ برنامه نویسی پیشرفته  
پایده سازی نرم افزار To do \_ بخش دوم

**استاد:**

دکتر مرتضی یوسف صنعتی

در تمرین چهارم، برنامه to do را در قالب یک برنامه کنسول پیاده سازی کردید. حال وقت آن رسیده است که رابط گرافیکی را به آن اضافه کنید.

رابط گرافیکی (GUI: Graphical User Interface) مجموعه ای شامل المان های گرافیکی نظیر دکمه ها، آیکون ها و ... می باشد که این امکان را به کاربر می دهد تا با برنامه تعامل داشته باشد.



## پیاده سازی

رابط گرافیکی که پیاده سازی می کنید، باید قابلیت های زیر را دارا باشد:

- امکان اضافه کردن Task جدید
- امکان حذف کردن Task ها
- امکان ویرایش عنوان Task ها
- امکان تکمیل Task ها ( بدین معنی که Task انجام شده است )
- نمایش لیستی از Task های انجام شده و انجام نشده
- امکان کردن یک Task Favorite

## اضافه کردن Task جدید :

کاربر با استفاده از دکمه Add a task می تواند یک Task جدید ایجاد کند. سپس برنامه با دریافت عنوان Task، آن را به لیست Task ها اضافه می کند.

+ Add a task

## لیست تمامی Task ها :

پس از اضافه کردن Task ها، تمامی Task ها (انجام شده و انجام نشده) باید نمایش داده شوند.

☒ Percy Thomas Pink (1876-1941)  
Individuals to research • 0 of 3 •

☐ Finish translation for History  
Tasks

☐ Source belated Father's Day present  
Tasks

## تکمیل کردن Task :

کاربر بعد از انجام یک Task می تواند با کلیک بر روی دکمه ای، آن را تیک بزند. (بدین معنی که آن Task انجام شده است)

☒ Percy Thomas Pink (1876-1941)  
Individuals to research • 0 of 3 •

☐ Finish translation for History  
Tasks

☐ Source belated Father's Day present  
Tasks

## Favorite کردن یک Task :

کاربر می تواند با استفاده از دکمه ای، Task هایی را که به آن علاقه مند است یا برایش مهم است را ستاره دار کند. (به عبارتی آن را تبدیل به یک Favorite Task کند)

☐ Finish translation for History  
Tasks

☐ Source belated Father's Day present  
Tasks

## حذف کردن Task :

کاربر باید بتواند هر Task موجود در لیست Task ها را حذف کند.

## ویرایش عنوان Task :

کاربر باید بتواند عنوان هر Task موجود در لیست Task ها را ویرایش کند.

## از چه ابزاری برای گرافیک برنامه استفاده کنیم؟

ابزار های مختلفی نظیر کتابخانه SFML و GTKmm یا فریمورک Qt برای اینکار وجود دارند. در توسعه این برنامه می توانید از هر ابزاری که ترجیح می دهید استفاده کنید.

## ذخیره سازی Task ها در فایل :

تمامی Task هایی که در برنامه ایجاد می شوند، باید در فایل های ذخیره شوند تا در صورت بسته شدن برنامه، Task ها باقی بمانند. بعد از هر بار اجرای برنامه، تمامی Task ها از این فایل بارگیری می شوند. شیوه ذخیره سازی Task ها در فایل به صورت باینری یا متنی تفاوتی ندارد. هر کدام از روش های باینری یا متنی را که ترجیح می دهید استفاده کنید.

## استفاده از گیت :

با توجه به اینکه این تمرین به صورت تیمی می باشد، استفاده از یک سیستم کنترل نسخه (Version Control System) نظیر گیت (git) می تواند بسیار موثر واقع شود. در نتیجه از شما انتظار می رود که از گیت به عنوان یک سیستم کنترل نسخه در توسعه این برنامه استفاده کنید. هنگام تحویل این تمرین، گزارش commit های خود را نیز ارسال کنید. برای اینکار می توانید به راحتی دستور زیر را در ترمینال وارد کنید تا فایل های حاوی تمامی commit های شما برایتان ساخته شود:



```
git log > log.txt
```

## توجه:

دقت داشته باشید که به صورت جزئی و دقیق، کامیت های اعضای تیم بررسی می شود. اعضای تیم همگی باید دارای commit باشند.

همچنین سعی کنید برای commit های خود، عناوین معنا داری انتخاب کنید.

برای یادگیری گیت، می توانید از دوره آموزش ویدیویی رایگان "گیت با جادی" استفاده کنید:

## نکات تکمیلی :

- انجام این تمرین به صورت تیمی و در قالب تیم های دو یا سه نفره می باشد. (در انتخاب اعضای تیم مختارید)
- استفاده از گیت برای این تمرین اجباری است.
- کدنویسی خود را به صورت مازولار انجام دهید .
- سورس کد برنامه باید حاوی کامنت هایی مناسب برای توضیح توابع، کلاس ها و ... باشد.
- برای نام گذاری توابع، متغیر ها، کلاس ها و ... از نام های معنا دار استفاده کنید.
- ارسال تمرین توسط یک نفر از اعضای تیم کافی است. (نام و نام خانوادگی و شماره دانشجویی اعضای تیم را در گزارش کار، در کنار سورس کد خود قرار دهید )