



چگونه بهتر کد بنویسیم

پاییز ۱۳۹۵
محمد امین پایدار

موضوع

- در این ارائه نامه چندی از عادت های خوب برنامه نویسی را بررسی می کنیم.

“I’m not a great programmer; I’m just a good programmer with great habits.”

— Kent Beck

انتخاب اسم

- همیشه اسامی که برای متغیر ها تابع ها و کلاس ها انتخاب میکنید باید توصیفی باشند.
- این کار کد را خوانا تر می کند.
- اگر اسامی منطقی انتخاب شده باشند فهم منطق کد راحت تر است.

• برنامه ای که حجم یک اتاق را حساب میکند

```
int a, b, c, z;  
cin >> a >> b >> c;  
z = a * b * c;  
cout << z << endl;
```



```
int width, height, length, volume;  
volume = width * height * length;  
cout << volume << endl;
```



محدودیت های انتخاب اسم

- اسم متغیر تابع و کلاس **نمی تواند** :

- کیورد های اصلی زبان باشند : مثل ... , if
- با **علائم خاص و اعداد** شروع شوند.

```
int -number;  
int *number;  
int 2books;
```



انتخاب اسم

- در صورتی که اسامی از چند بخش تشکیل شده اند به سه روش می توانید نام گذاری کنید

```
int number_of_books;
```

```
int numberOfBooks;
```

```
int NumberOfBooks;
```

- که همگی صحیح هستند اما **روش اول** در بین برنامه نویسان **سی** بیشتر رایج است.

تعریف متغیر



از تعریف متغیر ها در هر جای برنامه خود داری کنید.



همه ی متغیر ها را در ابتدای برنامه تعریف کنید.

حذف کد های بیهوده

ممکن است پیش بیاید که یک کد را که قبلا نوشته اید کامنت کنید و در زیر آن یک کد دیگر بنویسید و یادتان برود که کد قبلی را پاک کنید

توصیه می شود همیشه کد های بیهوده را پاک کنید.
البته این بیشتر در برنامه های بزرگ پیش می آید.

کد خفن و کد خوانا

- اکثرا بین کد خفن و کد خوانا اشتباه میشود.
- شاید کسی یک کدی که دیگری در ۲۰ خط زده در یک خط بزند.
- آیا این کار نشان دهنده ی خفن بودن است ؟
- آیا کدی که زده شده قابل فهم هم خواهد بود ؟
- گاهی اوقات آن کد یک خطی هم قابل فهم است اما اکثرا نه.
- سعی کنید کد قابل فهم بزنید



کد خوانا

- همه می دانند که دیباگ کردن خیلی سخت تر از نوشتن برنامه از ابتدا است. پس اگر بدانید که روزی می خواهید کد خود را دیباگ کنید چگونه کد می زنید؟
- علاوه بر این ممکن است روزی کسی روی پروژه ای که شما کار می کرده اید کار کند.
- در این صورت است که نوشتن کد خوانا اهمیت خود را نشان می دهد.

اسپیس و تب (فاصله و چندفاصله)

- گذاشتن تعداد مناسب اسپیس و تب بین دستورات با قالب خاصی به خوانا تر شدن کد کمک میکند.
- در هر زبان شاید این قالب ها تفاوت داشته باشد.
- لزومی نیست که همه از یک قالب استفاده کنند.
- چیزی که مهم است استوار بودن در استفاده ی یک قالب است.
- اگر قالبی را انتخاب کردید همیشه به آن قالب کد بزنید.

اسپیس و تب

```
int width,height,length,volume;
```



```
int width, height, length, volume;
```



اسپیس و تب

```
if( age>=20 ) { isYoung = true; cout << "some thing"; }
```



```
if (age >= 20)
{
    isYoung = true;
    cout << "some thing";
}
```



کامنت گذاری

- برای متغیر ها هر کدام در یک خط یک کامنت بگذارید.
- قبل از توابع یک کامنت چند خطی گذاشته و نحوه ی کار تابع را توضیح دهید (JavaDoc)
- اگر جایی کدی نوشته اید که منطقش پیچیده است در کامنت آن منطق و الگوریتم را توضیح دهید.

ادامه دارد...