

کتابخانه‌ی iTextSharp، یا همان برگردان iText جاوا، انصافاً اینقدر حرف برای گفتن دارد که [یک کتاب 600 صفحه‌ای](#) برای آن چاپ شده است، اما ... در حین استفاده از آن مشکل زیر (که به شکل وسیعی در قسمت‌های مختلف آن وجود دارد) قابل هضم نیست:

یکی از مواردی را که در حین طراحی یک API خوب باید در نظر گرفت، کمک به استفاده کننده در عدم بکارگیری Magic numbers است. حالا این Magic numbers یعنی چی؟ برای مثال قطعه کد زیر را در نظر بگیرید:

```
new Font(baseFont, 10, 0, BaseColor.BLACK)
```

مقدار پارامتر 3 در اینجا بی‌معنا است، مگر اینکه به مستندات مربوطه مراجعه کنیم. نگارش بهبود یافته کد فوق به شرح زیر است:

```
new Font(baseFont, 10, Font.NORMAL, BaseColor.BLACK)
```

کمی بهتر شد، اینبار ثوابت به یک کلاس منتقل شده‌اند و به این ترتیب معنای مقدار بکارگرفته شده بدون نیاز به مستندات متد فوق قابل درک است. اما این روش هم (یعنی همان روش متداول iTextSharp) دو مشکل مهم دارد: استفاده کننده محدودیتی در بکارگیری مقادیر ندارد، چون آرگومان‌ها از نوع int معرفی شده‌اند. ممکن است اشتباهی رخ دهد. باز هم نیاز است به مستندات کتابخانه مراجعه کرد، زیرا نوع int هیچ نوع منوی intellisense خاصی را ظاهر نمی‌کند. راه درست، استفاده از enum است بجای یک کلاس ساده که فقط یک سری از ثوابت در آن تعریف شده‌اند. نوع int را باید با enum زیر جایگزین کرد (یا ... بهتر است اینگونه بشود در کتابخانه‌ی اصلی ... روزی!)

```
public enum PdfFontStyle
{
    Normal = 0,
    Bold = 1,
    Italic = 2,
    Underline = 4,
    Strikethru = 8,
    BoldItalic = Bold | Italic
}
```

اگر در طراحی آن از ابتدا این روش پی‌گرفته می‌شد، منوی intellisense تبدیل به بهترین مستند این کتابخانه می‌شد.