Importing Library PyPDF2 (Basic Library)

Used for extracting text from pdf's

Note: Cannot Extract Images

In [42]:
```
import PyPDF2
```

In [81]:
```
a = PyPDF2.PdfFileReader("progit.pdf","rb")
```

Brief Info of PDF

In [21]:
```
# information of author,title,etc
a.getDocumentInfo()
```

Out[21]:
```
{'/Title': 'Pro Git',
 '/Author': 'Scott Chacon, Ben Straub',
 '/Creator': 'Asciidoctor PDF 2.3.0, based on Prawn 2.4.0',
 '/Producer': 'Scott Chacon, Ben Straub',
 '/ModDate': "D:20220915213417+00'00'",
 '/CreationDate': "D:20220915213521+00'00'"}
```

In [5]:
```
# same thing but without new line
print(a.documentInfo)
```

```
{'/Title': 'Pro Git', '/Author': 'Scott Chacon, Ben Straub', '/Creator': 'Asciidocto
r PDF 2.3.0, based on Prawn 2.4.0', '/Producer': 'Scott Chacon, Ben Straub', '/ModDa
te': "D:20220915213417+00'00'", '/CreationDate': "D:20220915213521+00'00'"}
```

Total Number of Pages

In [6]:
```
# total pages in pdf
print(a.getNumPages())
```

```
505
```

Whether the file is Encrypted or Not (Returns Bool)

In [33]:
```
# whether the file is encrypted or not
a.getIsEncrypted()
```

Out[33]:
```
False
```

Extracting Text From Any Page According To Page Number

In [8]:
```
# extracting text from any page according to page number
print(a.getPage(2).extractText())
```

```
Table of Contents
Licence . . . . . .
. . . . . .
. . . .  1
Preface by Scott Chacon . . . . . .
```

. 2
Preface by Ben Straub . . . . . .
. 3
Dedications . . . . . .
. . . . . .
. . . 4
Contributors . . . . . .
. . . . . .
. . 5
Introduction . . . . . .
. . . . . .
. . 8
Getting Started . . . . . .
. . . . . .
10
About Version Control . . . . . .
. 10
A Short History of Git . . . . . .
. 14
What is Git? . . . . . .
. 1
4
The Command Line . . . . . .
. 18
Installing Git . . . . . .
. 1
8
First-Time Git Setup . . . . . .
. 21
Getting Help . . . . . .
. 2
4
Summary . . . . . .
. 2
5
Git Basics . . . . . .
. . . . . .
. . . 26
Getting a Git Repository . . . . . .
. 26
Recording Changes to the Repository . . . . . .
. 28
Viewing the Commit History . . . . . .
. 40
Undoing Things . . . . . .
. 46
Working with Remotes . . . . . .
. 50
Tagging . . . . . .
. . . . . .
. 55
Git Aliases . . . . . .
. . . . . .
60
Summary . . . . . .
. 6
2
Git Branching . . . . . .
. . . . . .
. 63
Branches in a Nutshell . . . . . .
. 63
Basic Branching and Merging . . . . . .
. 70
Branch Management . . . . . .
. 79
Branching Workflows . . . . . .
. 82
Remote Branches . . . . . .

. 85
Rebasing 95
Summary 104
Git on the Server 105
The Protocols 105
Getting Git on a Server 110
Generating Your SSH Public Key 112
Setting Up the Server 113
Git Daemon 116

Converting PDF to Text file

In [11]:
```
# converting pdf to text file
str = ""
for i in range(1,11):
    str += a.getPage(i).extractText()
with open("textPDf.txt","w",encoding ='utf-8') as f:
    f.write(str)
```

Storing First and Last Page in New PDF

In [84]:
```
# storing first page in page1 and last page in page_l
page1 = a.getPage(0)
page_l = a.getPage(504)
```

In [85]:
```
# opening same pdf in writing format
a = PyPDF2.PdfFileWriter()
```

In [86]:
```
a.addPage(page1)
a.addPage(page_l)
```

In [87]:
```
# storing the first page and last page of pdf in new pdf
with open("New1.pdf","wb") as output:
    a.write(output)
    output.close()
```

Try To Run It By Yourself ☺

Refer to GitHub For Code