1. In what modes should the PdfFileReader() and PdfFileWriter() File objects will be opened?

ANS: These files will be opened in binary mode., read binary (rb) for PdfFileREader() and write binary (wb) PdfFileWriter()

In [2]:

*##!pip install PyPDF2*

2. From a PdfFileReader object, how do you get a Page object for page 5?

ANS: Calling getPage(4) will return a Page object for page 5 since page 0 is the first page

In [7]:

**import** PyPDF2 **as** pdf

pdfFileObj **=** open("meetingminutes.pdf",'rb')

pdfReader **=** pdf**.**PdfFileReader(pdfFileObj)

pageObj **=** pdfReader**.**getPage(4)

pageObj**.**extractText()

Out[7]:

'BOARD MINUTES\n -4- MARCH 7, 2014\n 5.1.\n9 On motion of Dr. Miranti, seconded by Mr. Lee, \nthe Board \napproved\n, as a \nNotice of Intent, revisions to Bulletin 1566, \nPupil Progression Policies and \nProcedures\n: §503. Regular Placement\n, as presented by the LDE\n. 5.1.\n10 On motion of Dr. Miranti, seconded by Mr. Lee, \nthe Board \napproved\n, as a \nNotice of Intent, revisions to Bulletin 111, \nThe Louisiana School, District, \nand State Accountability System\n: §301. School Performance Score Goal; \n§303. Transition from Fall 2013 to Sp\nring 2015; §413. Dropout/Credit \nAccumulation Index Calculations; §517. Inclusion of Schools; §521. \nPairing/Sharing of Schools with Insufficient Test Data; §603. Determining \na Cohort for \na Graduation; §611. Documenting a Graduation Index; \n§1301. Reward Elig\nibility; §2301. Schools Requiring \nReconstitution/Alternate Governance Plans; §3101. Appeals/Waivers and \nData Certification Processes; §3301. Inclusion of New Schools; §3303. \nReconfigured Schools; §4101. Valid Data Considerations; §4301. \nInclusion of All Di\nstricts; and §4317. District Accountability Data \nCorrections\n, as presented by the LDE\n. Dr. Beebe was recorded as being opposed to the motion.\n 5.1.\n11 On motion of \nDr. Miranti,\n seconded by \nMr. Lee,\n the Board receive\nd the \nSummary of Public Comments and Age\nncy Response regarding revisions\n to Bulletin 135, \nHealth and Safety\n: Chapter 5. \n Injury Management \nProgram Rules for Serious Sports Injuries and Chapter 7: Glossary, and \ndirect\ned BESE staff to pr\noceed\n with the final adoption of the January 20, \n2014, Notic\ne of Intent regarding revisions to Bulletin 135, \nHealth and \nSafety\n, Chapters 5 and\n 7. Agenda\n Item 5.2.\n Administration and Finance Committee\n (Schedule 3)\n 5.2.\n1 On motion of \nMr. Guillot,\n seconded by \nMr. Lee\n, the Board \nreceived\n the \nreport on 8(g) monitor\ning visits conducted by Board staff.\n 5.2.\n2 On motion of Mr. Guillot, seconded by Mr. Lee, \nthe Board \nreceived\n the \nquarterly report from the LDE Director of Internal Audit.\n 5.2.\n3 On motion of Mr. Guillot, seconded by Mr. Lee, \nthe Board \nreceived\n the \nrepor\nt on LDE contracts of $50,000 and under approved by the State \nSuperintendent of Education.\n '

3. What PdfFileReader variable stores the number of pages in the PDF document?

ANS: The PdfFileReader.numPages variable stores an integer of the number of pages in the PdfFileReader object

In [4]:

**import** PyPDF2 **as** pdf

pdfFileObj **=** open("meetingminutes.pdf",'rb')

pdfReader **=** pdf**.**PdfFileReader(pdfFileObj)

pdfReader**.**numPages

Out[4]:

19

4. If a PdfFileReader object’s PDF is encrypted with the password swordfish, what must you do before you can obtain Page objects from it?

ANS: before we obtain the page object, the pdf has to be decrypted by calling .decrypt('swordfish')

In [10]:

*#pdfReader.decrypt('swordfish')*

5. What methods do you use to rotate a page?

ANS: The rotateClockwise() and rotateCounterClockwise() methods. The degrees to rotate is passed as an integer argument

In [13]:

*#pageObj.rotateClockwise(180)*

6. What is the difference between a Run object and a Paragraph object?

ANS: Paragraph Object : A document contains multiple paragraphs. A paragraph begins on a new line and contains multiple

runs.

The Document object contains a list of Paragraph objects for the paragraphs in the document. (A new paragraph begins

whenever the user presses ENTER or RETURN while typing in a Word document.)

Run Objects :

Runs are contiguous groups of characters within a paragraph with the same style

7. How do you obtain a list of Paragraph objects for a Document object that’s stored in a variable named doc?

ANS: By using doc.paragraphs

In [23]:

*#!pip install python-docx*

**import** docx

doc **=** docx**.**Document('demo.docx')

doc**.**paragraphs

Out[23]:

[<docx.text.paragraph.Paragraph at 0x28cce43d100>,

<docx.text.paragraph.Paragraph at 0x28cce43de20>,

<docx.text.paragraph.Paragraph at 0x28cce554f70>,

<docx.text.paragraph.Paragraph at 0x28cce554f10>,

<docx.text.paragraph.Paragraph at 0x28cce554b50>,

<docx.text.paragraph.Paragraph at 0x28cce642c40>,

<docx.text.paragraph.Paragraph at 0x28cce5b6bb0>]

8. What type of object has bold, underline, italic, strike, and outline variables?

ANS: A Run object has bold, underline,italic,strike and outline variables

9. What is the difference between False, True, and None for the bold variable?

ANS: Runs can be further styled using text attributes. Each attribute can be set to one of three values:

True (the attribute is always enabled, no matter what other styles are applied to the run),

False (the attribute is always disabled),

None (defaults to whatever the run’s style is set to)

True always makes the Run object bolded and False makes it always not bolded, no matter what the style’s bold setting is. None will make the Run object just use the style’s bold setting

10. How do you create a Document object for a new Word document?

ANS: By Calling the docx.Document() function.

11. How do you add a paragraph with the text 'Hello, there!' to a Document object stored in a variable named doc?

ANS: **import** docx

doc **=** docx**.**Document()

doc**.**add\_paragraph('Hello there!')

doc**.**save('hellothere.docx')

12. What integers represent the levels of headings available in Word documents?

ANS: integer from 0 to 4

The arguments to add\_heading() are a string of the heading text and an integer from 0 to 4. The integer 0 makes the heading the Title style, which is used for the top of the document. Integers 1 to 4 are for various heading levels, with 1 being the main heading and 4 the lowest subheading