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

Ans: PdfFileReader() and PdfFileWriter() objects in Python's PyPDF2 library should be opened in binary mode ('rb' for read and 'wb' for write) when working with PDF files.

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

Ans: page\_number = 4 # Page numbers are 0-based

page = pdf\_reader.getPage(page\_number)

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

Ans: pdf\_reader.numPages

**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: If a PdfFileReader object's PDF is encrypted with the password "swordfish," you must decrypt it using the provided password before you can obtain Page objects. You can use the following code:

pdf\_reader.decrypt('swordfish')

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

Ans: To rotate a page in a PDF using PyPDF2, you can use the rotateClockwise() and rotateCounterClockwise() methods on a Page object.

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

Ans: In the context of the python-docx library for working with Word documents, a Run object represents a contiguous run of text with the same style within a paragraph. A Paragraph object represents a single paragraph of text in a document.

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

Ans: To obtain a list of Paragraph objects from a Document object named doc, you can use the following code: paragraphs = doc.paragraphs

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

Ans: The Run object in the python-docx library has bold, underline, italic, strike, and outline variables.

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

Ans: In the context of python-docx, the bold variable can take the following values:

True: The text in the Run object will be bold.

False: The text in the Run object will not be bold.

None: The bold setting for the Run object will not be changed.

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

Ans: To create a Document object for a new Word document using the python-docx library, you can use the following code:

from docx import Document

doc = Document()

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

Ans: To add a paragraph with the text 'Hello, there!' to a Document object stored in a variable named doc, you can use the following code: doc.add\_paragraph('Hello, there!')

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

Ans: 0: Title

1: Heading 1

2: Heading 2

3: Heading 3

4: Heading 4

5: Heading 5

6: Heading 6