```python
"""
CS241 Checkpoint 6A
Written by Chad Macbeth
"""

### Represents any book
class Book:

    ### Initialize a book object
    def __init__(self):
        self.title = ""
        self.author = ""
        self.publication_year = 0

    ### Get book data from user
    def prompt_book_info(self):
        self.title = input("Title: ")
        self.author = input("Author: ")
        self.publication_year = int(input("Publication Year: "))

    ### Display book data
    def display_book_info(self):
        print("{} ({}) by {}" .format(self.title, self.publication_year, self.author))

### TextBook IS-A Book.  This means that the TextBook will have all the same
### data and functions that exist in the Book Class.  THe goal here is reuse!
class TextBook(Book):

    ### Initialize a TextBook object
    def __init__(self):
        super().__init__()   # When you do inheritance, the first thing your __init__
                             # function should do is call the __init__ function in the
                             # base class.  The super() function is used to call functions
                             # in the base class

        self.subject = ""    # After initializing all the data in the Book class, we can
                             # now initialize data unique to the TextBook class

    ### Get Textbook specific information from the user
    def prompt_subject(self):
        self.subject = input("Subject: ")

    ### Display Textbook specific information
    def display_subject(self):
        print("Subject: {}" .format(self.subject))

### PictureBook IS-A Book.  It also has illustrator information.
class PictureBook(Book):

    ### Initialize a PictureBook object
    def __init__(self):
        super().__init__()
        self.illustrator = ""

    ### Get Picturebook specific information from the user
    def prompt_illustrator(self):
        self.illustrator = input("Illustrator: ")

    ### Display Picturebook specific information
    def display_illustrator(self):
        print("Illustrated by {}" .format(self.illustrator))

### Test the classes created above
def main():
    book = Book()
    book.prompt_book_info()
```

```python
67         print()
68         book.display_book_info()
69         print()
70
71         text = TextBook()
72         text.prompt_book_info()
73         text.prompt_subject()
74         print()
75         text.display_book_info()
76         text.display_subject()
77         print()
78
79         picture = PictureBook()
80         picture.prompt_book_info()
81         picture.prompt_illustrator()   # If you tried to do text.prompt_illustrator()
82                                        # it would not work.  TextBook objects don't have
83                                        # a prompt_illustrator function.
84         print()
85         picture.display_book_info()
86         picture.display_illustrator()
87
88
89  if __name__ == "__main__":
90         main()
91
```