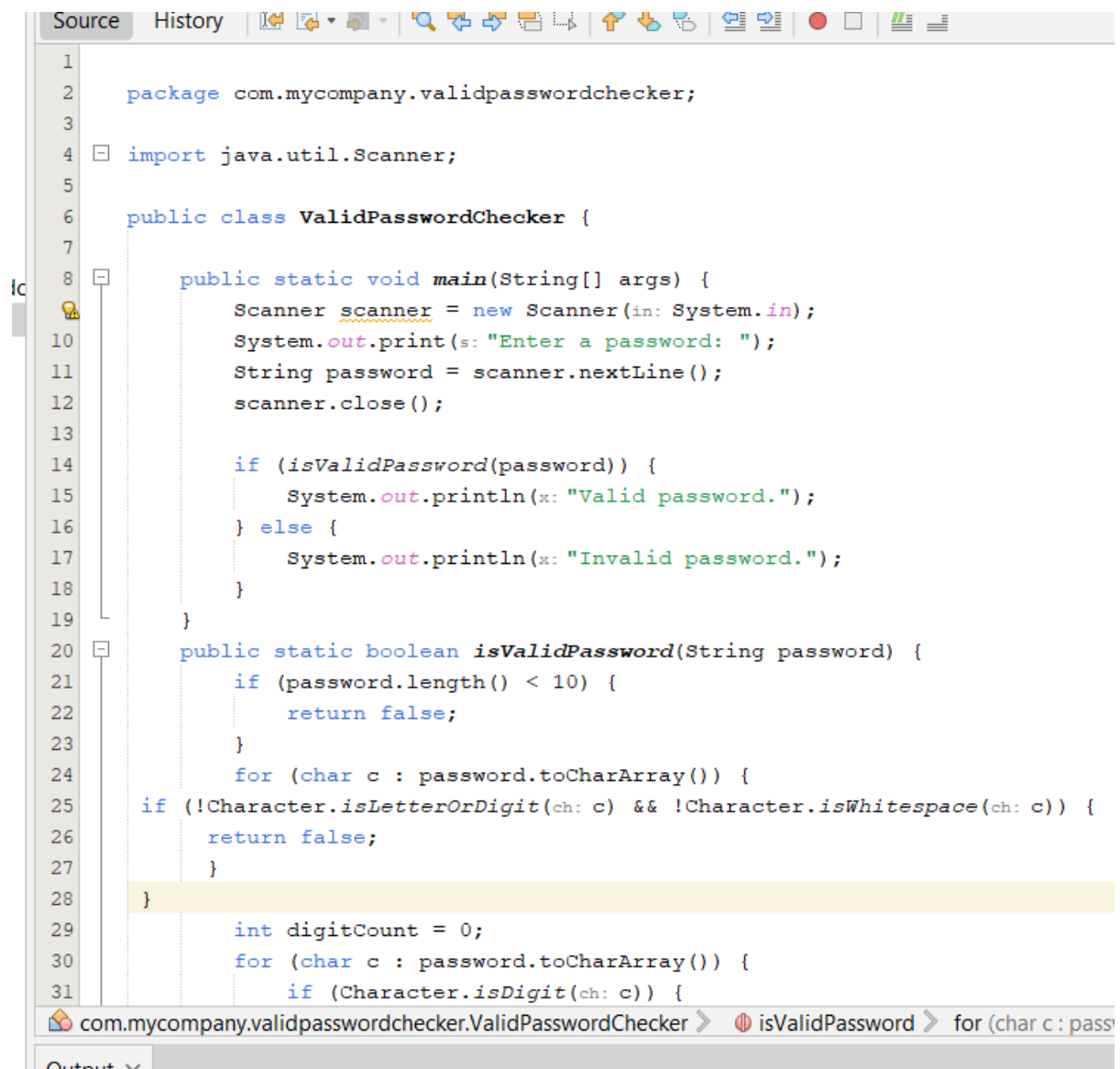


First Task: Write a Java program that checks whether a string is a valid password. Suppose the password rule is as follows:

- A password must have at least ten characters.
- A password consists of only letters and digits.
- A password must contain at least two digits.
- A password must contain a whitespace.

(You have to take input of the password from the user).

Source Code:

The image shows a screenshot of a Java IDE with a source code editor. The code is for a class named 'ValidPasswordChecker' in the package 'com.mycompany.validpasswordchecker'. It imports 'java.util.Scanner'. The 'main' method uses a 'Scanner' to take input from the user, prints 'Enter a password:', and then checks if the password is valid using the 'isValidPassword' method. The 'isValidPassword' method contains logic to check the password length (must be at least 10), if it contains only letters and digits, and if it contains at least two digits. The code is partially highlighted in yellow. The IDE interface includes a toolbar at the top and a breadcrumb trail at the bottom: 'com.mycompany.validpasswordchecker.ValidPasswordChecker > isValidPassword > for (char c : pass'.

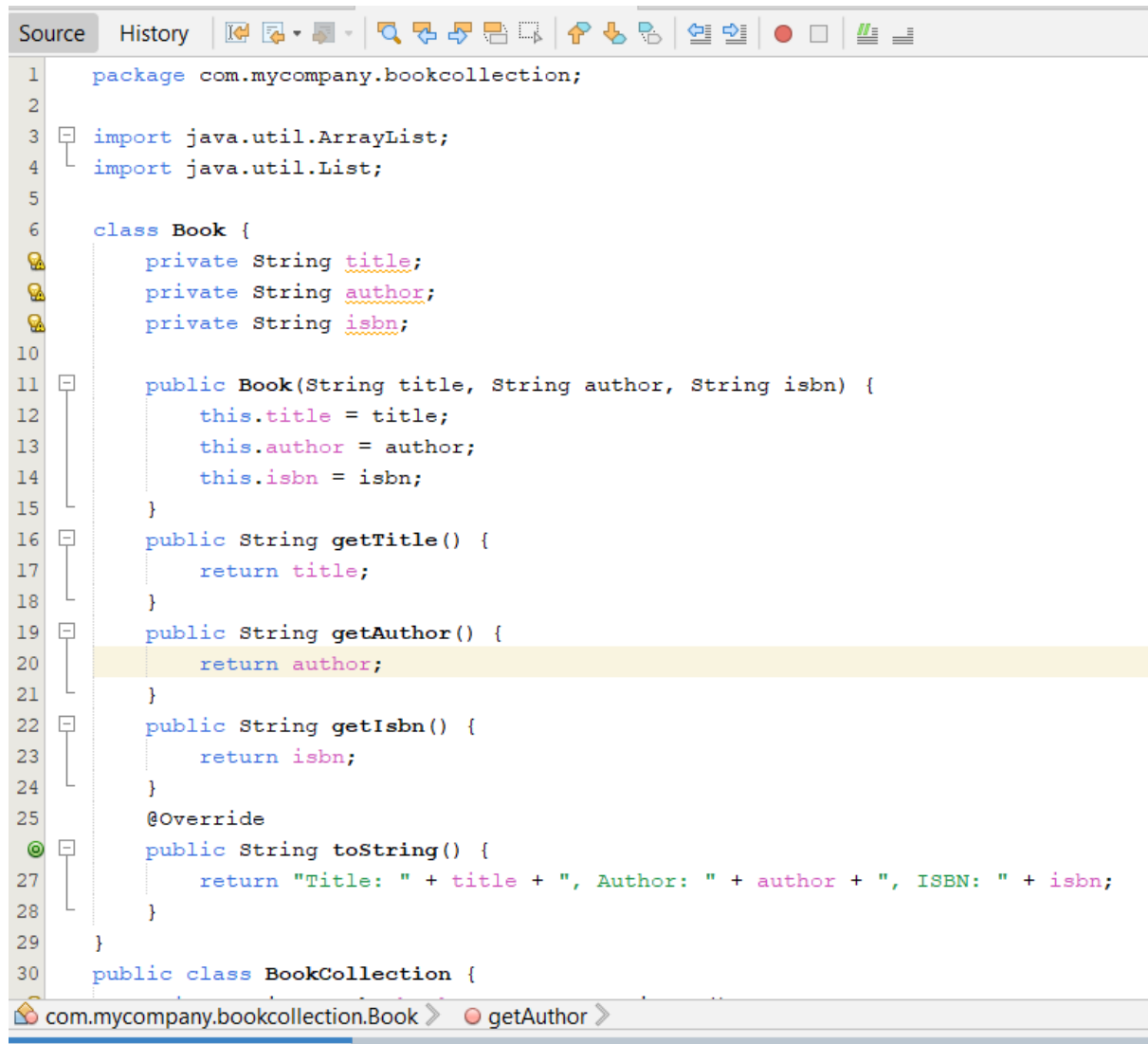
```
1
2 package com.mycompany.validpasswordchecker;
3
4 import java.util.Scanner;
5
6 public class ValidPasswordChecker {
7
8     public static void main(String[] args) {
9         Scanner scanner = new Scanner(System.in);
10        System.out.print(s: "Enter a password: ");
11        String password = scanner.nextLine();
12        scanner.close();
13
14        if (isValidPassword(password)) {
15            System.out.println(x: "Valid password.");
16        } else {
17            System.out.println(x: "Invalid password.");
18        }
19    }
20
21    public static boolean isValidPassword(String password) {
22        if (password.length() < 10) {
23            return false;
24        }
25        for (char c : password.toCharArray()) {
26            if (!Character.isLetterOrDigit(ch: c) && !Character.isWhitespace(ch: c)) {
27                return false;
28            }
29
30            int digitCount = 0;
31            for (char c : password.toCharArray()) {
32                if (Character.isDigit(ch: c)) {
```

Output:

Discussion: Here is a Java program that checks whether a given string is a valid password or not. This program takes input from the user and checks whether the entered password meets the specified rules. It uses a combination of loops and conditional statements to validate the password according to the given criteria.

Second Task: Write a Java program to create a class called "Book" with attributes for title, author, and ISBN, and methods to add and remove books from a collection.

Source Code:



```
1  package com.mycompany.bookcollection;
2
3  import java.util.ArrayList;
4  import java.util.List;
5
6  class Book {
7      private String title;
8      private String author;
9      private String isbn;
10
11      public Book(String title, String author, String isbn) {
12          this.title = title;
13          this.author = author;
14          this.isbn = isbn;
15      }
16      public String getTitle() {
17          return title;
18      }
19      public String getAuthor() {
20          return author;
21      }
22      public String getIsbn() {
23          return isbn;
24      }
25      @Override
26      public String toString() {
27          return "Title: " + title + ", Author: " + author + ", ISBN: " + isbn;
28      }
29  }
30  public class BookCollection {
```

The screenshot shows an IDE window with a toolbar at the top. The code is written in Java. The 'Book' class has three private attributes: title, author, and isbn. It has a constructor and three getter methods. The 'BookCollection' class is partially visible at the bottom. The IDE's status bar at the bottom shows the current file path as 'com.mycompany.bookcollection.Book' and the current method as 'getAuthor'.

```
Source History
30 public class BookCollection {
31     private List<Book> books = new ArrayList<>();
32     public void addBook(Book book) {
33         books.add(book);
34     }
35     public void removeBook(Book book) {
36         books.remove(book);
37     }
38     public List<Book> getBooks() {
39         return books;
40     }
41     public static void main(String[] args) {
42         BookCollection collection = new BookCollection();
43         collection.addBook(new Book(title: "The Cow", author: "Gorge Henri Cow", isbn: "027-147389"));
44         collection.addBook(new Book(title: "Hamba Hamba", author: "Robert H. Hamba", isbn: "027-873629"));
45         collection.addBook(new Book(title: "Siuuu", author: "Christiano Ronaldo", isbn: "027-753895"));
46         List<Book> books = collection.getBooks();
47         System.out.println("Books in the collection:");
48         for (Book book : books) {
49             System.out.println(book);
50         }
51         Book bookToRemove = new Book(title: "The Cow", author: "Gorge Henri Cow", isbn: "027-14738");
52         collection.removeBook(bookToRemove);
53
54         books = collection.getBooks();
55         System.out.println("\nBooks in the updated collection:");
56         for (Book book : books) {
57             System.out.println(book);
58         }
59     }
60 }
```

Output:

```
Output - Run (BookCollection) x
-----< com.mycompany:BookCollection >-----
Building BookCollection 1.0-SNAPSHOT
from pom.xml
-----[ jar ]-----
--- resources:3.3.0:resources (default-resources) @ BookCollection ---
skip non existing resourceDirectory C:\Users\HP\OneDrive\Documents\NetBeansPr
--- compiler:3.10.1:compile (default-compile) @ BookCollection ---
Changes detected - recompiling the module!
Compiling 1 source file to C:\Users\HP\OneDrive\Documents\NetBeansProjects\Bc
--- exec:3.1.0:exec (default-cli) @ BookCollection ---
Books in the collection:
Title: The Cow, Author: Gorge Henri Cow, ISBN: 027-147389
Title: Hamba Hamba, Author: Robert H. Hamba, ISBN: 027-873629
Title: Siuuu, Author: Christiano Ronaldo, ISBN: 027-753895

Books in the updated collection:
Title: The Cow, Author: Gorge Henri Cow, ISBN: 027-147389
Title: Hamba Hamba, Author: Robert H. Hamba, ISBN: 027-873629
Title: Siuuu, Author: Christiano Ronaldo, ISBN: 027-753895
-----
BUILD SUCCESS
-----
Total time: 1.130 s
```

Discussion: Here's a Java program that defines a Book class with attributes for title, author, and ISBN, and includes methods to add and remove books from a collection. In this program, we have a Book class with attributes for title, author, and ISBN, along with getters to access these attributes. The BookCollection class manages a collection of books using a List<Book>. It provides methods to add and remove books from the collection. Finally, in the main method, we demonstrate how to create a collection, add books to it, remove a book, and display the contents of the collection.