LAB-PROGRAM-3

Create a class Book which contains four members : name, author, price, num-pages. Includes a constructor to set values for the members. Include methods to set & get the details of the objects. Include toString() method that could display the complete details of the book. Develop a Java program to create n book objects.

```java
import java.util.Scanner;

class Book {
    String name;
    String author;
    double price;
    int num-pages;
    Book (String name, String author, double price, int numpages)
    {  this.name = name;
       this.author = author;
       this.price = price;
       this.num-pages = num-pages;
    }
    void setDetails (String name, String author, double price,
    int num-pages) {  this.name = name;
            this.author = author;
            this.price = price;
            this.num-pages = num-pages;
    }
    void getDetails () {
    String s = "Book : " + name + "by author:" + author +
         "with pages :" + num-pages + "is of price :" + price;
    System.out.println(s);
}}
```

```java
public String toString() {
    String s = "Book : " + name + " by author : " + au
        + " with pages : " + num_pages + " is of p
        + price ;
    return s;
}
}


public class Main {
public static void main ( String [] args) {
    Scanner scanner = new Scanner (System
    System.out.println (" Enter the no. of books to cre");
    int n = scanner.nextInt ();
    Book [] books = new Book [n];

    for ( int i = 0; i < n ; i++) {
        scanner.nextLine ();
        System.out.println ("\n Enter details for Book " + (i + ":\n");   Boo
        System.out.print ("Name : ");                                      cons
        String name = scanner.nextLine ();
        System.out.print (" Author : ");
        String author = scanner.nextLine ();
        System.out.print (" Price :");
        double price = scanner.nextDouble ();
        System.out.print ("Number of Pages : ");
        int numPages = scanner.nextInt ();
        books [i] = new Book (name, author, price, numPages);
    }

    System.out.println ("\n Details of the books :");
    for ( int i = 0; i < n ; i++) {
        System.out.println ("\n Book " + (i + 1) + "\n" + books[i].toString());
    }
}
scanner.close ();
}
}
```
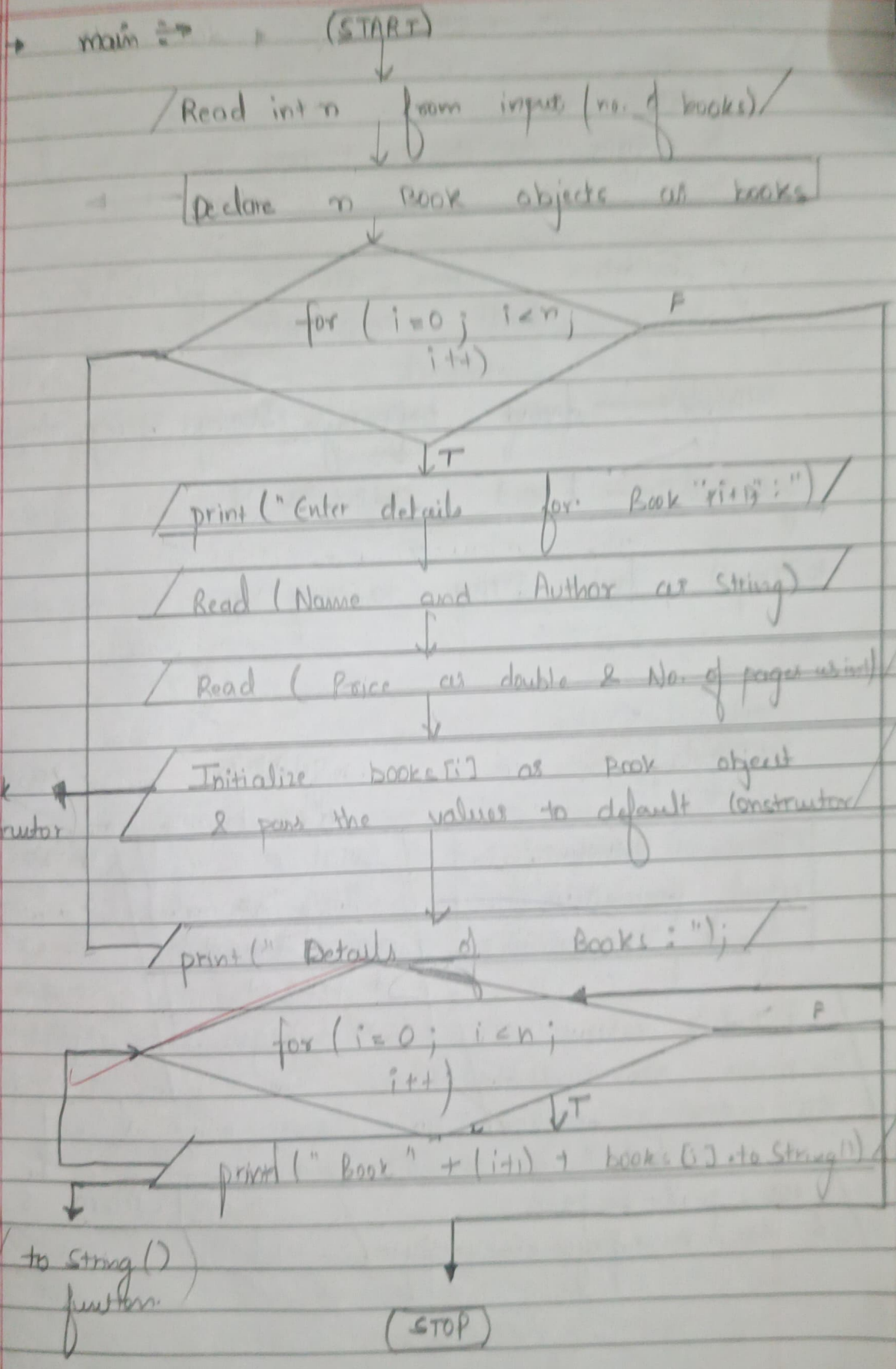
main ≈ →   (START)

Read int n   from input (no. of books)

Declare   n   Book   objects   as   books

for (i=0 ; i<n;   F
              i++)

                          ↓ T

print (" Enter details   for: Book "i+1" : ")

Read (Name   and   Author   as String)

Read (Price   as   double & No. of pages as int)

Initialize   books[i]   as   Book   object
& pass the   values   to default Constructor

"n);/ Book
Constructor

print (" Details   of   Books : ");

for (i=0 ; i<n;   F
              i++)
                          ↓ T

print (" Book " + (i+1) + books [i] .to String ")

to String ()
function.

(STOP)

Class Book

Declare name and author as String, price as double and num_pages as int

Initialize name, author, price & num_pages using default constructor

setDetails ——— | void   setDetails (String, String, double, ...)

getDetails () ← | void   getDetails ()

toString () | public String   toString ()

void
setDetails function
↓

```
• Receive details from
  actual parameters to
  formal parameters
```
↓
```
Initialize value
this.name = name
this.author = author
```
↓
```
this.price = price
& this.num_pages = num_pages
```

getDetails
↓
```
Initialize
String s =
name + author
+ price +
num_pages
```

toString
↓
```
Initialize s=
s = "name" +
name + "author"
+ author + "price"
+ price + "pages"
+ num_pages
```
↓
```
return s
```