Q01.

package q01;

public class **main** {

public static void main(String[] args) {

Factorial fobj = new Factorial();

System.out.println(fobj.Factorial(5));

}

}

package q01;

public class **Factorial** {

public int Factorial(int n){

if (n > 1) {

return n \* Factorial(n - 1);

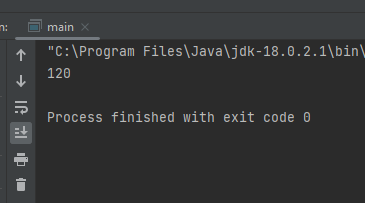
} else {

return 1;

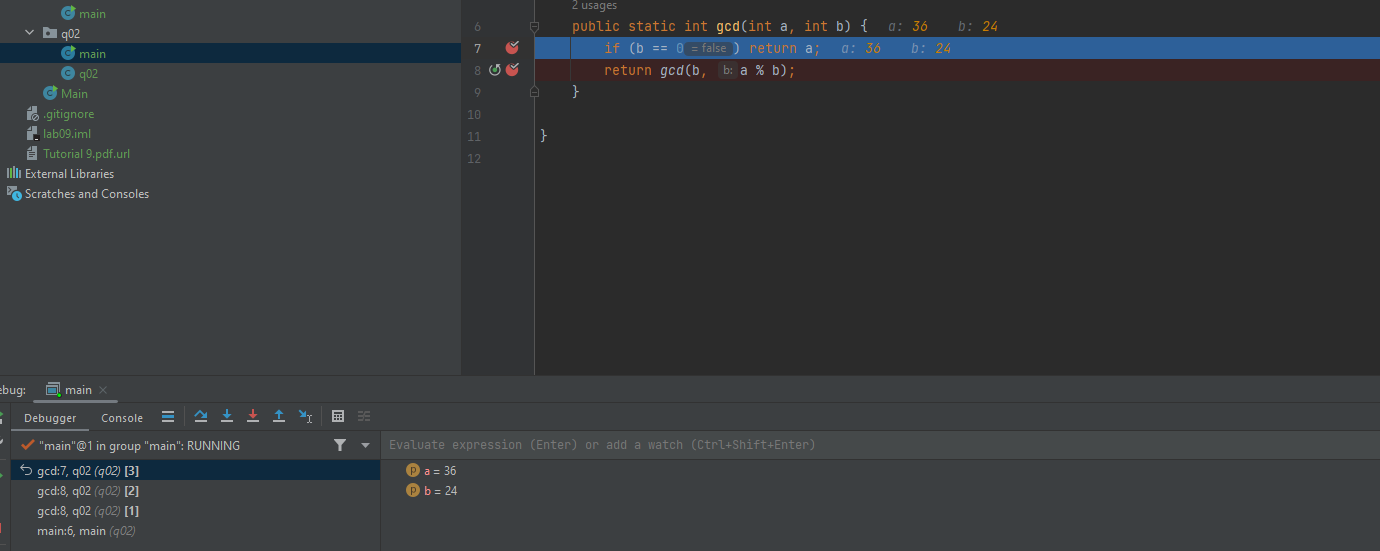
}

}

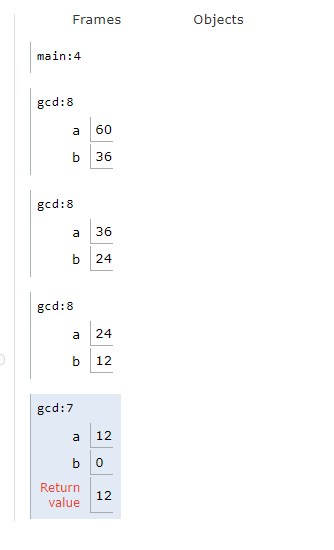
}



Q02.



When we call the gcd(a, b) method in main(), we first assign the two values, a and b. Then, the method checks if the condition b == 0 is true. If it is, it returns a. If not, it recursively calls the gcd function with the arguments (b, a % b).

For example gcd(60, 36):

* The first call is gcd(60, 36). Since 36 != 0, the method moves to the next call: gcd(36, 60 % 36).
* Now, 60 % 36 equals 24, so the method continues with gcd(36, 24).
* For gcd(36, 24), since 24 != 0, it calls gcd(24, 36 % 24).
* Now, 36 % 24 equals 12, so the method proceeds with gcd(24, 12).
* For gcd(24, 12), since 12 != 0, it calls gcd(12, 24 % 12).
* Now, 24 % 12 equals 0, so the method calls gcd(12, 0).

At this point, since b == 0, the method returns the value of a, which is 12.

So, the greatest common divisor (GCD) of 60 and 36 is 12.

Q03.

package q03;

public class **main** {

public static void main(String[] args) {

fibbonachi fibObj = new fibbonachi();

System.out.println(fibObj.Fibbonachi(12));

}

}

package q03;

public class **fibbonachi** {

public int Fibbonachi(int i) {

if (i == 0) {

return 0;

} else if (i == 1) {

return 1;

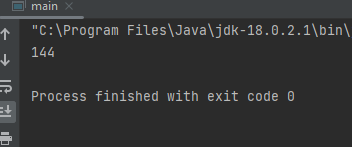
} else {

return Fibbonachi(i-1) + Fibbonachi(i-2);

}

}

}



Q04.

package q04;

public class **TestAccount** {

public static void main(String[] args) {

Account account1 = new Account("A013", "Alice", 1500);

Account account2 = new Account("A014", "Bob", 500);

System.out.println("Initial Balances:");

account1.displayInfo();

account2.displayInfo();

System.out.println("Transferring 300 from Alice to Bob...");

account1.transferTo(account2, 300);

account1.displayInfo();

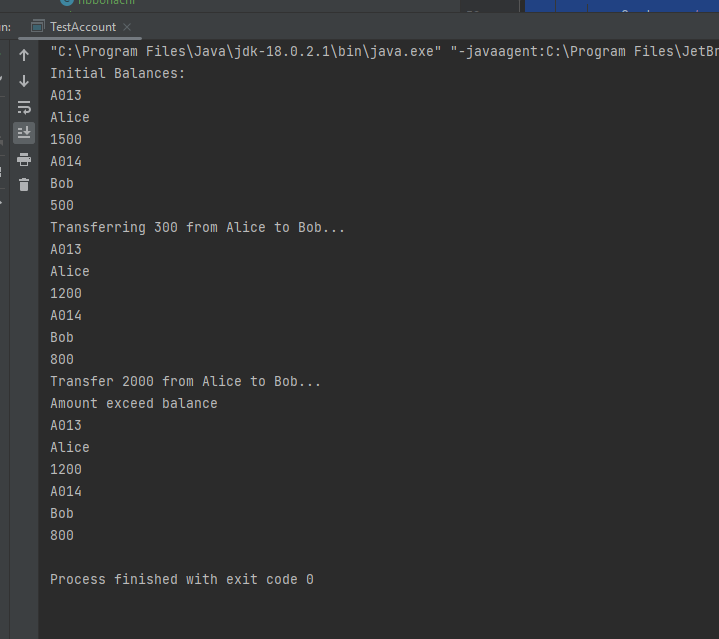
account2.displayInfo();

System.out.println("Transfer 2000 from Alice to Bob...");

account1.transferTo(account2, 2000);

account1.displayInfo();

account2.displayInfo();

 }

}

package q04;

public class **Account** {

private String name;

private int balance = 0;

private String id;

public int credit(int balance) {

this.balance += balance;

return this.balance;

}

public String getId() {

return id;

}

public int debit(int ammount) {

if (this.balance >=ammount){

this.balance -= ammount;

} else {

System.out.println( "Amount exceed balance");

}

return ammount;

}

public int transferTo(Account account,int ammount){

if (this.balance >=ammount){

account.credit(ammount);

this.balance -= ammount;

} else {

System.out.println( "Amount exceed balance");

}

return ammount;

}

public void displayInfo() {

System.out.println(id);

System.out.println(name);

System.out.println(balance);

}

public int getBalance() {

return balance;

}

public String getName() {

return name;

}

public Account(String id, String name, int balance) {

this.id = id;

this.name = name;

this.balance = balance;

}

public Account(String id, String name) {

this.id = id;

this.name = name;

}

}

Q05.

package q05;

public class **GymMemberClient** {

public static void main(String[] args) {

GymMember member1 = new GymMember("John Doe", 10);

System.out.println("Booking 3 sessions...");

member1.bookSession();

member1.bookSession();

member1.bookSession();

member1.displayMemberInfo();

member1.cancelSession();

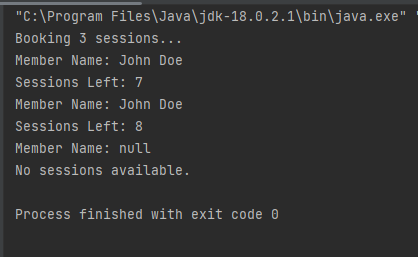
member1.displayMemberInfo();

GymMember member2 = new GymMember();

member2.displayMemberInfo();

}

}



package q05;

public class **GymMember** {

private String memberName;

private int sessionsLeft;

public GymMember() {

this.memberName = null;

this.sessionsLeft = 0;

}

public GymMember(String memberName, int sessionsLeft) {

this.memberName = memberName;

this.sessionsLeft = sessionsLeft;

}

public String getMemberName() {

return memberName;

}

public int getSessionsLeft() {

return sessionsLeft;

}

public void bookSession() {

if (sessionsLeft > 0) {

sessionsLeft--;

} else {

System.out.println("No sessions left to book.");

}

}

public void cancelSession() {

sessionsLeft++;

}

public void displayMemberInfo() {

if (sessionsLeft > 0) {

System.out.println("Member Name: " + memberName);

System.out.println("Sessions Left: " + sessionsLeft);

} else {

System.out.println("Member Name: " + memberName);

System.out.println("No sessions available.");

}

}

}

Q06.