Functions 2

Agenda

1. Function Examples
2. Given N return sum of all even number
3. Given radius find area of circle
4. Even or odd
5. Given M check if it is a perfect square

Problem 1

Given N find and return sum of all even numbers till N ?

Example 1

```
N = 10
```

Solution

```
N = 10

Even numbers less than equal to 10

2 -> 4 -> 6 -> 8 -> 10

adding 2 + 4 + 6 + 8 + 10 = 30
```

Example 2

```
N = 5
```

Solution

```
N = 5

Even numbers less than equal to 5

2 -> 4

adding 2 + 4 = 6
```

Approach

- Iterate from 2 to N with an increament of 2 in each step of iteration
- Add the values obtained in each iteration.

Code

```
 static int evenSum(int N){
    int sum = 0;
    for(int i = 2 ;i <= N ; i += 2){
        sum += i;
    }
    return sum;
}
public static void main(String[] args) {
    Scanner scanner = new Scanner(System.in);
    int N = scanner.nextInt();
    int sum = evenSum(N);
    System.out.println(sum);
}
```

Dry run

```
N = 7
```

Step	i	i <= N	Sum = Sum + i	i = i + 2
1	2	Yes	0+2 = 2	4
2	4	Yes	2+4 = 6	6
3	6	Yes	6+6 = 12	8
4	8	No		

Problem 2

Given R (radius of the circle) find area of the circle

Approach

- Calculate area using formula π*r^2
- But do we know the exact value of π ?
- We might think it to be 3.14 but it is just an approximation.
- therefore we will use the value of π using Math.PI in java.

Code

```
  static double areaOfCircle(int R) {
    double area = Math.PI * R * R;
    return area;
}

public static void main(String[] args) {
    Scanner scanner = new Scanner(System.in);
    int radius = scanner.nextInt();
    double circleArea = areaOfCircle(radius);
    System.out.println(circleArea);
}
```

Function Rules

- **When will a function end?**
    - When all lines are executed.
    - We execute return statement in function.
- **What will happen a function ends ?**
    - We will go back to line from where it was called.

Question

What will be the output?

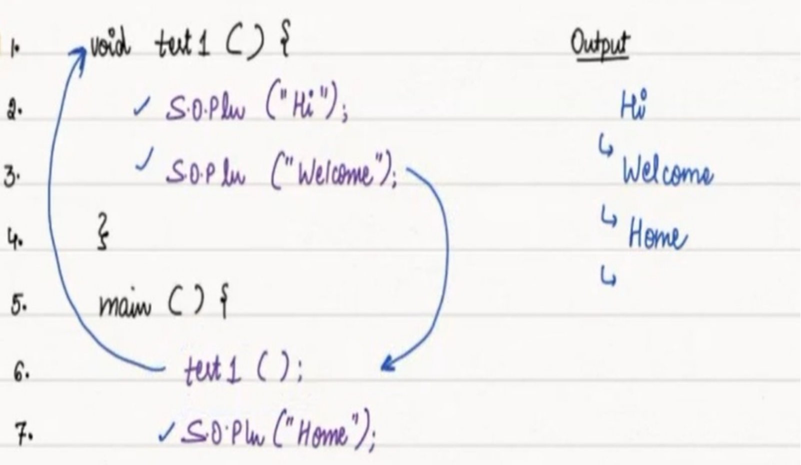
```
public static void test1(){
    System.out.println("Hi");
    System.out.println("Welcome");
}
public static void main(){
    test1();
    System.out.println("Home");
}
```

Choices

- ☑ Hi
  Welcome
  Home
- □ Welcome
  Home
  Hi
- □ Home
  Welcome
  Hi

Explaination

- According to rule `main()` is executed and first `test1()` gets called.
- Inside test1 there is no return statement since it is void. test ends after executing all lines of code so Hi and Welcome are printed.
- after test1 ends we go back again in `main()` and Home is printed



Question

What will be the output?

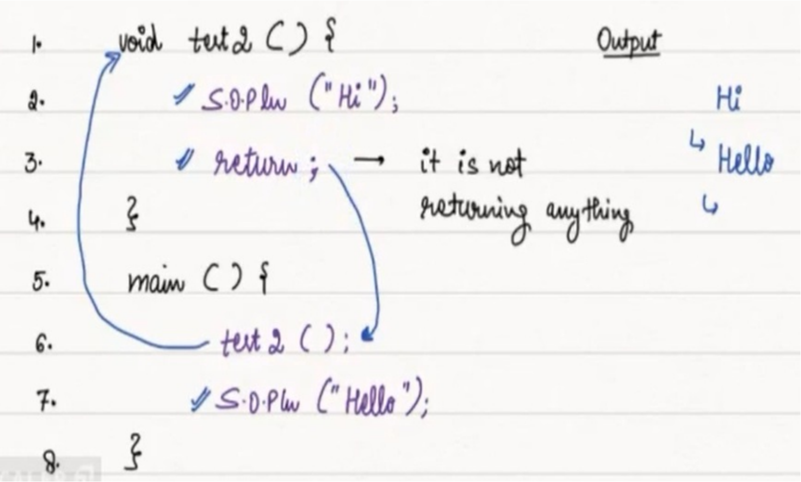
```
public static void test2(){
    System.out.println("Hi");
    return;
}
public static void main(){
    test2();
    System.out.println("Hello");
}
```

Choices

- ☑ Hi
  Hello
- ☐ Hello
  Hi
- ☐ Error

Explaination

- Some people might answer Error because of return statement in void type function But will it actually produce an error ?
- Since return statement is not returning any thing it won't produce an error.



```
void test2 () {
    ✓ S.O.Pln ("Hi");
    ✓ return ;
}
main () {
    test 2 ();
    ✓ S.O.Pln ("Hello");
}
```

Question

What will be the output?

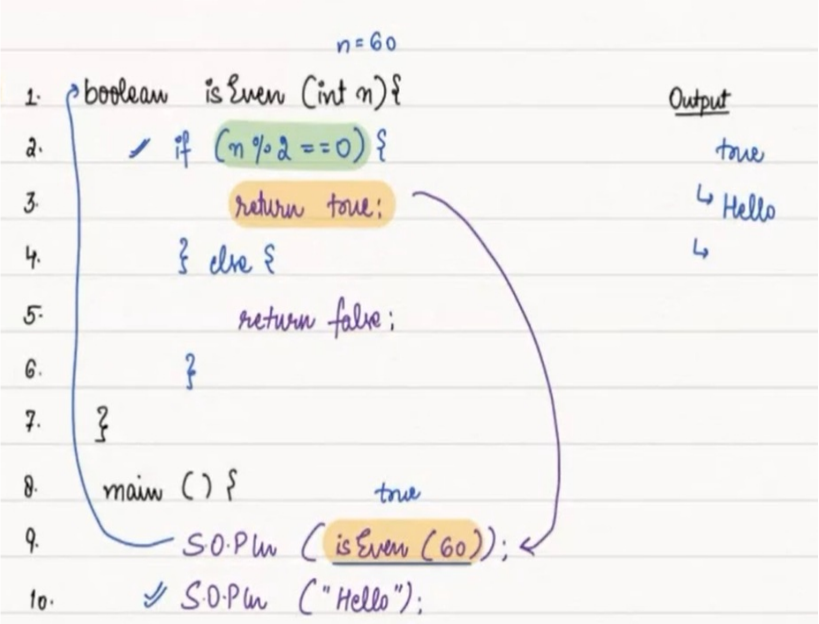
```
public static boolean isEven(int N){
    if(N % 2 == 0){
        return true;
    }else{
        return false;
    }
}
public static void main(){
    System.out.println(isEven(60));
    System.out.println("Hello");
}
```

Choices

- ☑ true
  Hello
- ☐ Hello
  false
- ☐ false
  Hello

Explaination

- We are calling isEven function with n = 60. So on entering the function we check for if n%2 == 0
- Since 60 % 2 ==0 it enters if statement and returns true and isEven execution ends here
- After that Hello is printed


n=60
Output
true
↳ Hello
↳
true

```
boolean isEven (int n){
    if (n%2 ==0){
        return true;
    } else {
        return false;
    }
}
main () {
    S.O.Pln (isEven (60));
    S.O.Pln ("Hello");
```

Question

What will be the output?

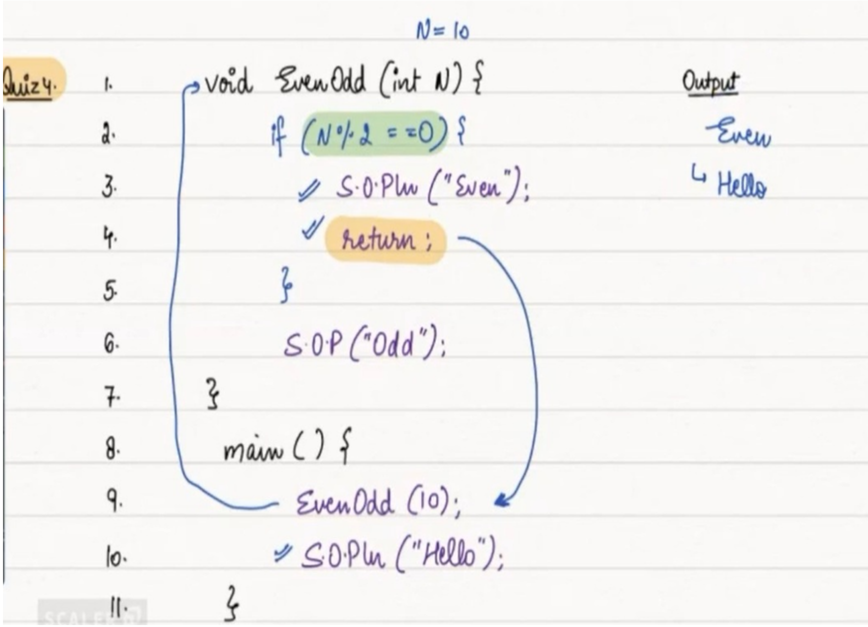
```
public static boolean EvenOdd(int N){
    if(N % 2 == 0){
        System.out.println("Even");
        return;
    }
    System.out.print("Odd");
}
public static void main(){
    EvenOdd(10);
    System.out.println("Hello");
}
```

Choices

- ☑ Even
  Hello
- ☐ Odd Hello
- ☐ Hello

Explaination

- We are calling EvenOdd function with N = 10 So on entering function the we check for if N%2 == 0
- Since 10 % 2 ==0 it enters if statement an prints Even and due to return statement EvenOdd execution ends.
- After that Hello is printed.



Quiz4.

N= 10

```
void EvenOdd (int N) {
    if (N%2 ==0) {
        S.O.Pln ("Even");
        return ;
    }
    S.O.P ("Odd");
}
main () {
    EvenOdd (10);
    S.O.Pln ("Hello");
}
```

Output

```
Even
↳ Hello
```

Question

What will be the output?

```
 public static int check(int N){
    System.out.print(N+10);
}
public static void main(){
    check(15);
}
```

Choices

- ☐ 15
- ☑ Error
- ☐ No output

Explaination

- When we run the function we get an ERROR : Missing return statement in function

> Note : If any function having return type other then void then should have atleast one return statement.

Question

What will br the output?

```
 public static int even(int n){
    if(n % 2 == 0){
        return 2;
    }
}
public static void main(){
    int a = even(10);
    System.out.print(a);
}
```

Choices

- ☐ Even
- ☑ Error
- ☐ 2

Explanation

- When we run the function we get an ERROR : Missing return statement in function
- This occured because compiler checks if the condition is false then is there any return statement. Since there is no return statement Error is thrown.
- **If you create a function and there is a mistake in it. It doesn't weather you call it or not you will get an error**
- **If any function having return type other then void then should have return statements for all cases.**
- So According to rules if we add return statement after if we can resolve the error

Question

What will be the output?

```
 public static int test(int n){
    if(n % 2 == 0){
        return 2;
    }
    if(n % 5 == 0){
        return 5;
    }
}
public static void main(){
    int a = test(15);
}
```

Choices

- ☐ 5
- ☑ Error
- ☐ 2
- ☐ No output

Explaination

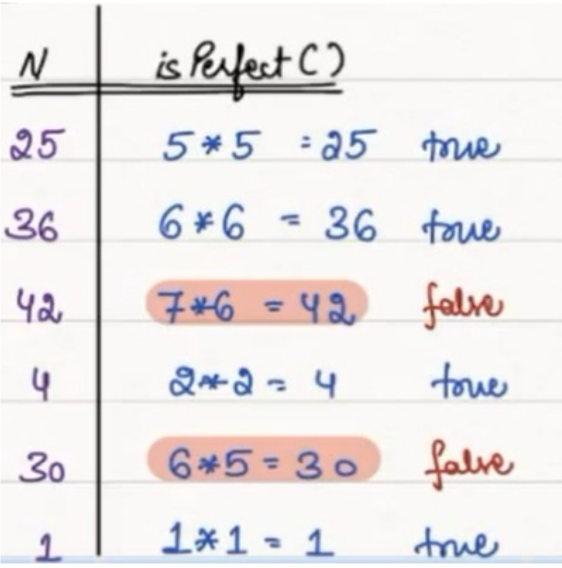
- When we run the function we get an ERROR : Missing return statement in function
- This occured because compiler checks if the first condition is false then is there any return statement.
- Compiler checks if the second condition is false then is there any return statement.
- Since there is no return statement Error is thrown.

Problem 3

Given N return true if the number is perfect square else return false?

> N is said to be a perfect square if it can be expresses as product of two equal positive numbers.

Testcases

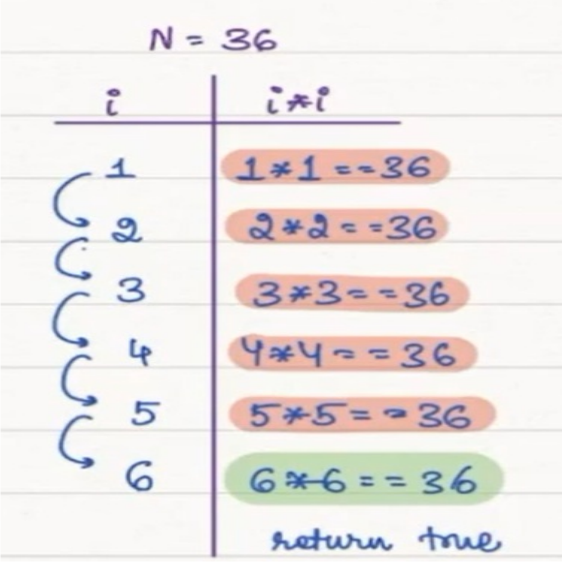


N	isPerfect()	
25	5*5 = 25	true
36	6*6 = 36	true
42	7*6 = 42	false
4	2*2 = 4	true
30	6*5 = 30	false
1	1*1 = 1	true

Idea

- Starting from 1 (Smallest +ve number) check if there is any number p that satisfies the condition **p*p == N**

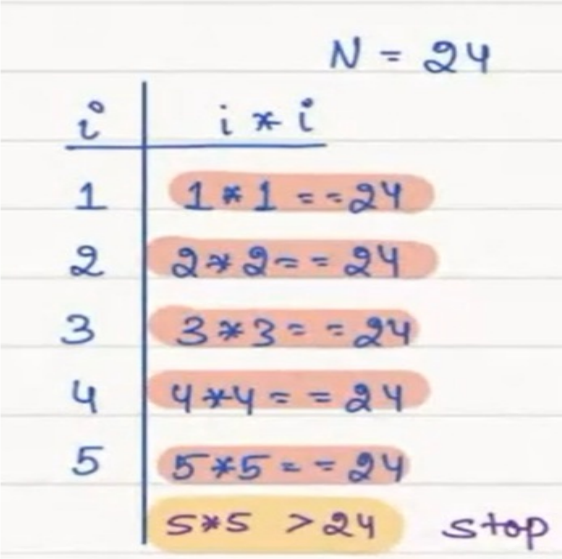
Example 1



N = 36

i	i*i
1	1*1 == 36
2	2*2 == 36
3	3*3 == 36
4	4*4 == 36
5	5*5 == 36
6	6*6 == 36
	return true

Example 2



N = 24

i	i*i	
1	1*1 == 24	
2	2*2 == 24	
3	3*3 == 24	
4	4*4 == 24	
5	5*5 == 24	
	5*5 > 24	stop

Observations

- We can observe from above examples that :-
    - We need to iterate starting from i=1
    - We need iterate till i*i <= N
    - On each iteration we check if i*i == N if so then return true else we return false

Code

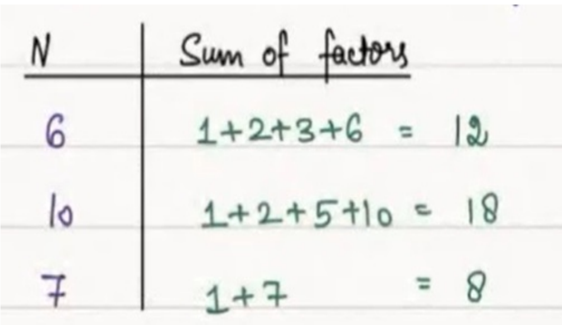
```
static boolean isPerfectSquare(int N){
    int i = 1;
    while(i * i <= N)
    {
      if(i * i == N)
      {
        return true;
      }
      i ++ ;
    }
    return false;
}
public static void main(String[] args) {

    Scanner scanner = new Scanner(System.in);
    int N = scanner.nextInt();
    System.out.println(isPerfectSquare(N));
}
```

Problem 4

Given N find the sum of all factor of N ?

Testcases



N	Sum of factors
6	1+2+3+6 = 12
10	1+2+5+10 = 18
7	1+7 = 8

Observations & Approach

- All factors of a number lies in range [1,N].
- We iterate over range [1,N] and if a number is an factor we add it to the answer.

Code

```
static int sumOfFactors(int N){
    int ans = 0;
    for(int i = 1;i <= N;i ++ ){
        if(N % i == 0){
          ans += i;
        }
    }
    return ans;
}
public static void main(String[] args) {
    Scanner scanner = new Scanner(System.in);
    int N = scanner.nextInt();
    System.out.println(sumOfFactors(N));
}
```

Question

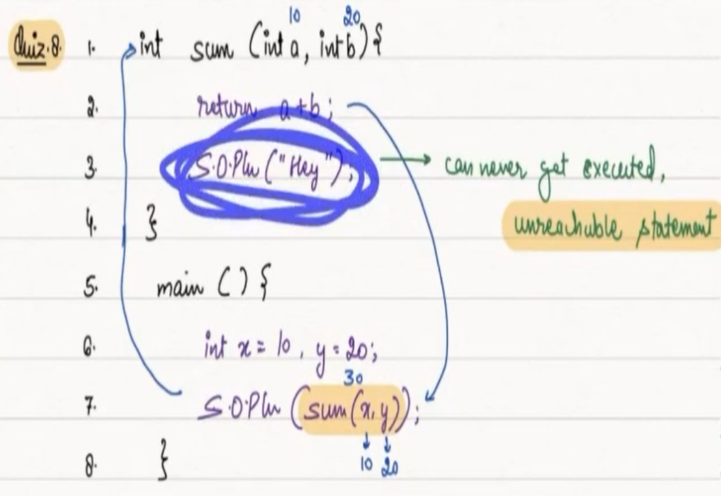
What will be the output?

```
 public static int sum(int a, int b){
     return a+b;
     System.out.println("Hey");
}
public static void main(){
    int x = 10, y = 20;
    System.out.pritnln(sum(x,y));
}
```

Choices

- ☑ Error
- ☐ 30
- ☐ Hey

Explanation



```
int sum (int a, int b){
    return a+b;
    S.O.Pln ("Hey");
}
main ( ) {
    int x = 10, y = 20;
    S.O.Pln (sum(x,y));
}
```

can never get executed, unreachable statement

- Here error is generated because of print command just after return in function sum
- The code after return can not be executed still compiler sees valid statements(not commented lines) and hence throws unreachable statement error

- Statements after return are not executed. If there are statements after return then compiler throws error --> [Statement Unreachable]

Question

What will be the output?

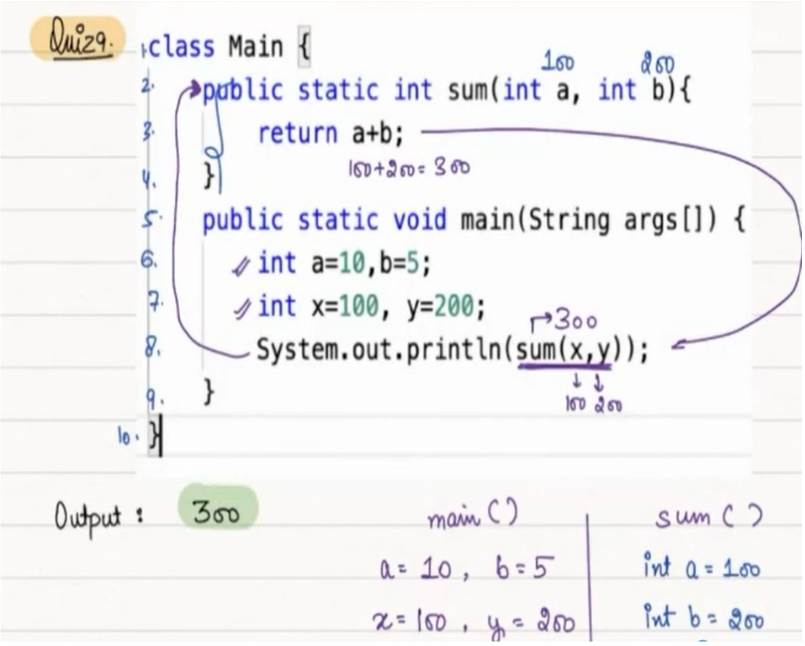
```
 public static int sum(int a, int b){
     return a+b;
}
public static void main(){
    int a = 10, b = 5;
    int x = 100, y = 200;
    System.out.pritnln(sum(x,y));
}
```

Choices

- ☑ 300
- ☐ 15

- □ Compilation error

Explaination



```
class Main {
public static int sum(int a, int b){
    return a+b;
}
public static void main(String args[]) {
    int a=10,b=5;
    int x=100, y=200;
    System.out.println(sum(x,y));
}
}
```

- We get answer 300 because value passed to function sum was 100 and 200 respectively(i.e. values of x and y of main function) hence value of variables a and b in function sum is set to 100 and 200 respectively and value of variables a and b in function sum is independent of the values of a and b defined in function main
- **Therefore we can say that variables in a function are defined only in scope of the function**

Question

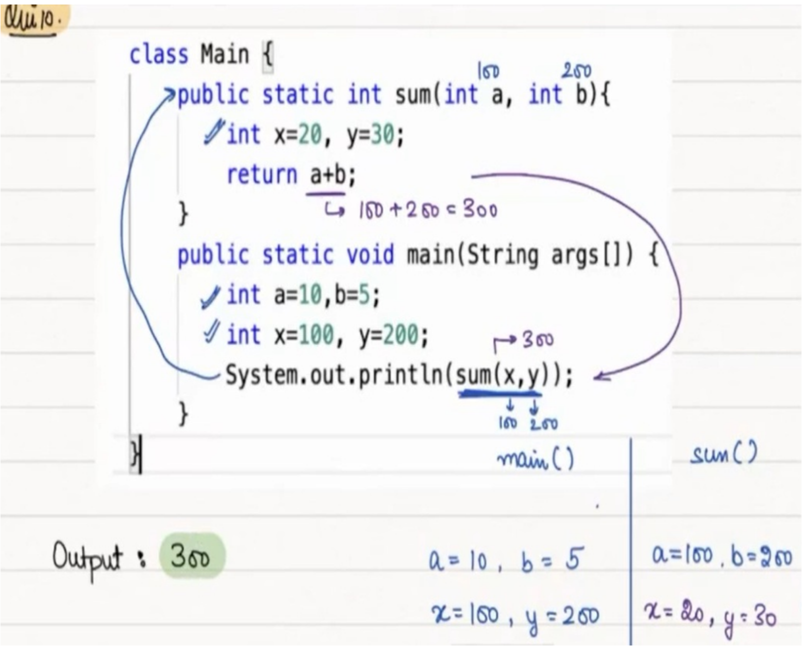
What will be the output?

```
 public static int sum(int a, int b){
     int x = 20, y = 30;
     return a+b;
}
public static void main(){
     int a = 10, b = 5;
     int x = 100, y = 200;
     System.out.pritnln(sum(x,y));
}
```

Choices

- ☑ 300
- □ 15
- □ Compilation error

Explaination



```
class Main {
    public static int sum(int a, int b){
        int x=20, y=30;
        return a+b;
    }
    public static void main(String args[]) {
        int a=10,b=5;
        int x=100, y=200;
        System.out.println(sum(x,y));
    }
}
```

- We get answer 300 because value passed to function sum was 100 and 200 respectively(i.e. values of x and y of main function) hence value of variables a and b in function sum is set to 100 and 200 respectively and value of variables a and b in function sum is independent of the values of a and b defined in function main
- Also x and y defined in `sum()` & x and y defined in `main()` are seperate entities and changing the value of x and y in `sum()` won't affect the x and y defined in `main()` .

Question

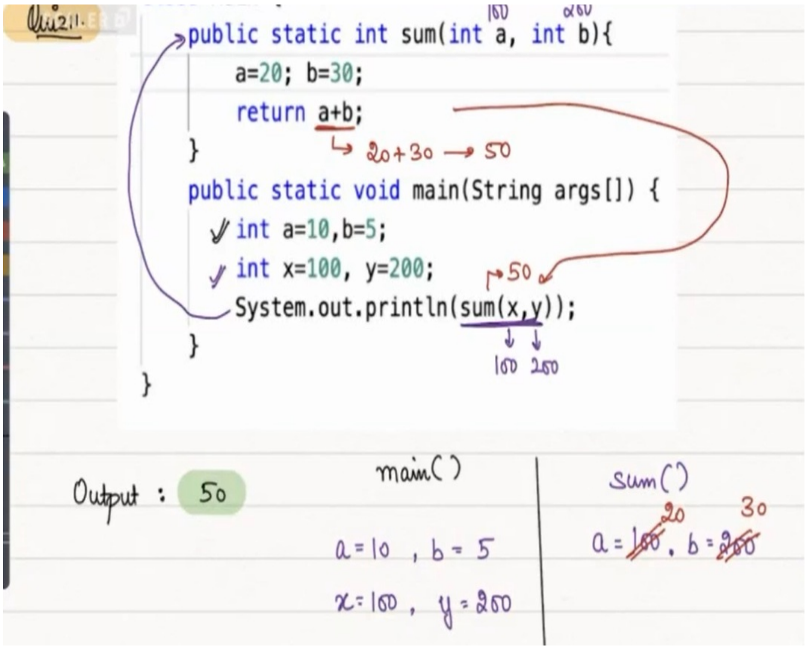
What will be the output?

```
 public static int sum(int a, int b){
     a = 20; b = 30;
     return a+b;
}
public static void main(){
    int a = 10, b = 5;
    int x = 100, y = 200;
    System.out.pritnln(sum(x,y));
}
```

Choices

- ☑ 50
- ☐ 300
- ☐ Compilation error

Explaination



```
public static int sum(int a, int b){
    a=20; b=30;
    return a+b;
}
public static void main(String args[]) {
    int a=10,b=5;
    int x=100, y=200;
    System.out.println(sum(x,y));
}
}
```

- We get answer 50 because value passed to function sum was 100 and 200 respectively(i.e. values of x and y of main function) hence value of variables a and b in function sum is set to 100 and 200 respectively and value of variables a and b in function sum is independent of the values of a and b defined in function main
- But in `sum()` we set the value of a and b to 20 and 30 respectively since a and b are defined within scope of `sum()` there values are changed.

Question

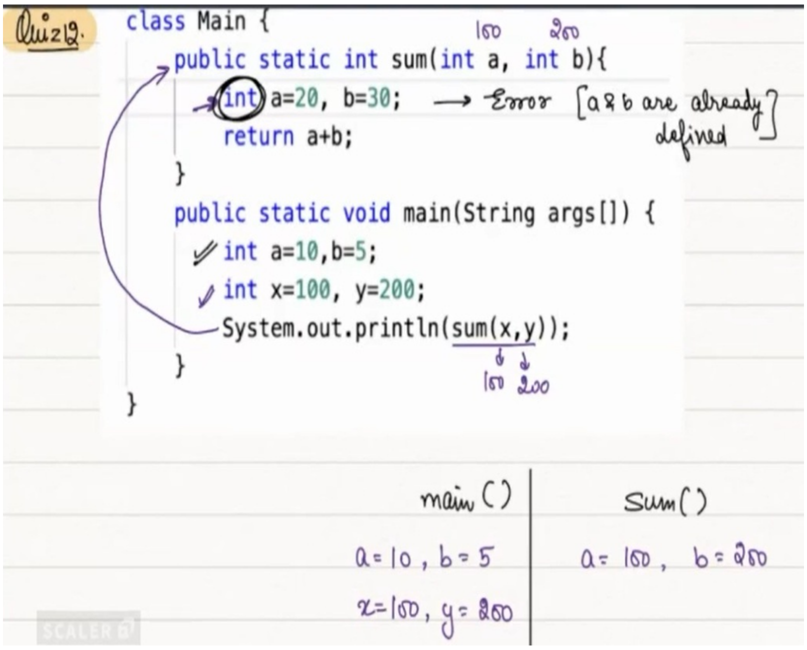
What will be the output?

```
 public static int sum(int a, int b){
    int a = 20, b = 30;
    return a+b;
}
public static void main(){
    int a = 10, b = 5;
    int x = 100, y = 200;
    System.out.pritnln(sum(x,y));
}
```

Choices

- □ 50
- □ 300
- ☑ Error

Explaination



Quiz12.

```
class Main {
    public static int sum(int a, int b){
        int a=20, b=30;
        return a+b;
    }
    public static void main(String args[]) {
        int a=10,b=5;
        int x=100, y=200;
        System.out.println(sum(x,y));
    }
}
```

main()	sum()
a=10, b=5	a= 100, b= 200
x=100, y= 200	

- We get compilation error because we declared variables that are already defined in function `sum()`