Intermediate Java Questions with Solutions and Explanations

1. Sum of Even Numbers

```
public class SumEven {
    public static void main(String[] args) {
        int sum = 0;
        for (int i = 2; i <= 100; i += 2) {
            sum += i;
        }
        System.out.println("Sum of even numbers from 1 to 100: " + sum);
    }
}
// Explanation:
// - Initialize sum to 0
// - Loop through even numbers using i += 2
// - Add each number to sum
```

2. Factorial Using Function

```
public class FactorialFunction {
    public static int factorial(int n) {
        int result = 1;
        for (int i = 1; i <= n; i++) {
            result *= i;
        }
        return result;
    }
    public static void main(String[] args) {
        int num = 5;
        System.out.println("Factorial of " + num + " is " + factorial(num));
    }
}
// Explanation:
// - Function multiplies all numbers from 1 to n
// - Called from main
```

3. Number Guessing Game

```
import java.util.Scanner;
public class GuessGame {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        int target = 7, guess;
        do {
            System.out.print("Guess a number between 1 and 10: ");
            guess = sc.nextInt();
            if (guess != target) {
                System.out.println("Wrong, try again!");
            }
        } while (guess != target);
        System.out.println("Correct!");
    }
}
```

```
// Explanation:
// - Uses do-while to repeat until correct guess
```

4. Check Prime Number

```
public class PrimeCheck {
    public static boolean isPrime(int n) {
        if (n <= 1) return false;
        for (int i = 2; i <= n / 2; i++) {
            if (n % i == 0) return false;
        }
        return true;
    }
    public static void main(String[] args) {
        int num = 29;
        System.out.println(num + " is prime? " + isPrime(num));
    }
}
// Explanation:
// - Prime has no divisors from 2 to n/2
```

5. Pattern Printing

```
public class StarPattern {
    public static void main(String[] args) {
        for (int i = 1; i <= 5; i++) {
            for (int j = 1; j <= i; j++) {
                System.out.print("*");
            }
            System.out.println();
        }
    }
}
// Explanation:
// - Outer loop controls rows
// - Inner loop prints * i times
```

6. Find Maximum of Three Numbers

```
public class MaxOfThree {
    public static int max(int a, int b, int c) {
        if (a >= b && a >= c)
            return a;
        else if (b >= c)
            return b;
        else
            return c;
    }
    public static void main(String[] args) {
        System.out.println("Max: " + max(10, 20, 15));
    }
}
// Explanation:
// - Compares three numbers using if-else
```

7. Count Digits in a Number

```
import java.util.Scanner;
public class DigitCounter {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        System.out.print("Enter a number: ");
        int num = sc.nextInt(), count = 0;
        while (num != 0) {
            num /= 10;
            count++;
        }
        System.out.println("Number of digits: " + count);
    }
}
// Explanation:
// - Remove last digit until number is 0
```

8. Multiplication Table

```
public class TablePrinter {
    public static void printTable(int n) {
        for (int i = 1; i <= 10; i++) {
            System.out.println(n + " x " + i + " = " + (n * i));
        }
    }
    public static void main(String[] args) {
        printTable(5);
    }
}
// Explanation:
// - Print n times 1 to 10 in loop
```

9. Reverse a Number

```
public class ReverseNumber {
    public static void main(String[] args) {
        int num = 1234, reversed = 0;
        while (num != 0) {
            int digit = num % 10;
            reversed = reversed * 10 + digit;
            num /= 10;
        }
        System.out.println("Reversed number: " + reversed);
    }
}
// Explanation:
// - Extract last digit and build reversed
```

10. Check Palindrome Number

```
public class PalindromeCheck {
    public static boolean isPalindrome(int num) {
        int original = num, reversed = 0;
        while (num != 0) {
            int digit = num % 10;
            reversed = reversed * 10 + digit;
            num /= 10;
        }
        return original == reversed;
    }
    public static void main(String[] args) {
        int num = 121;
        System.out.println(num + " is palindrome? " + isPalindrome(num));
    }
}
// Explanation:
// - Reverse number and compare with original
```