SCALA PROGRAMMING
LAB - 13

NAME - KAPAROTU VENKATA SURYA THARANI
USN - 22BTRAD018
BRANCH - AI & DE

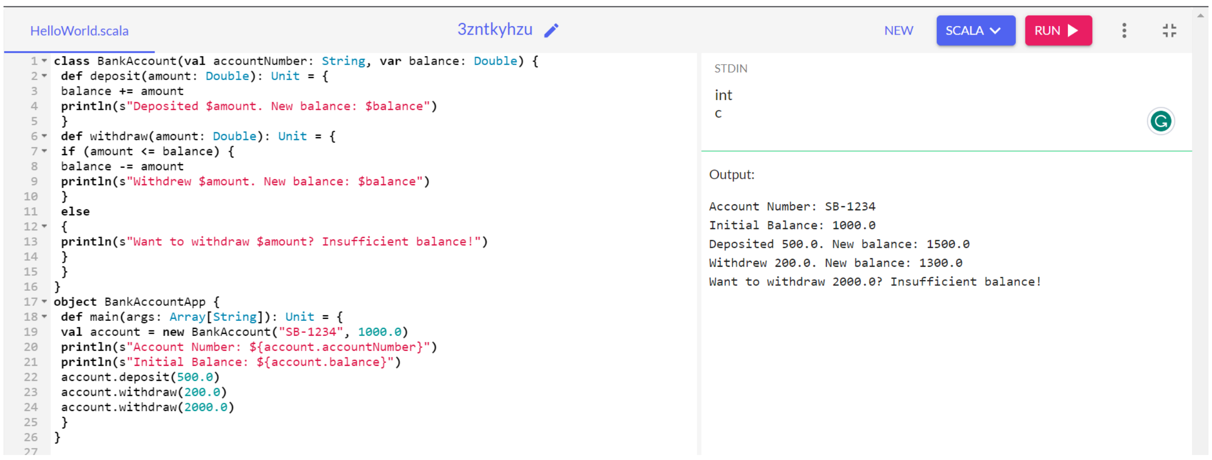
Ques - Write a Scala program that creates a class BankAccount with properties accountNumber and balance. Implement methods to deposit and withdraw money from the account.

CODE :

```
        class BankAccount(val accountNumber: String, var balance: Double) {
 def deposit(amount: Double): Unit = {
 balance += amount
 println(s"Deposited $amount. New balance: $balance")
 }
 def withdraw(amount: Double): Unit = {
 if (amount <= balance) {
 balance -= amount
 println(s"Withdrew $amount. New balance: $balance")
 }
 else
 {
 println(s"Want to withdraw $amount? Insufficient balance!")
 }
 }
}
object BankAccountApp {
 def main(args: Array[String]): Unit = {
 val account = new BankAccount("SB-1234", 1000.0)
 println(s"Account Number: ${account.accountNumber}")
 println(s"Initial Balance: ${account.balance}")
 account.deposit(500.0)
 account.withdraw(200.0)
 account.withdraw(2000.0)
 }
}
```

OUTPUT :

Account Number: SB-1234
Initial Balance: 1000.0
Deposited 500.0. New balance: 1500.0
Withdrew 200.0. New balance: 1300.0
Want to withdraw 2000.0? Insufficient balance!



```scala
class BankAccount(val accountNumber: String, var balance: Double) {
 def deposit(amount: Double): Unit = {
 balance += amount
 println(s"Deposited $amount. New balance: $balance")
 }
 def withdraw(amount: Double): Unit = {
 if (amount <= balance) {
 balance -= amount
 println(s"Withdrew $amount. New balance: $balance")
 }
 else
 {
 println(s"Want to withdraw $amount? Insufficient balance!")
 }
 }
}
object BankAccountApp {
 def main(args: Array[String]): Unit = {
 val account = new BankAccount("SB-1234", 1000.0)
 println(s"Account Number: ${account.accountNumber}")
 println(s"Initial Balance: ${account.balance}")
 account.deposit(500.0)
 account.withdraw(200.0)
 account.withdraw(2000.0)
 }
}
```

```
int
c
```

Output:

```
Account Number: SB-1234
Initial Balance: 1000.0
Deposited 500.0. New balance: 1500.0
Withdrew 200.0. New balance: 1300.0
Want to withdraw 2000.0? Insufficient balance!
```

In this program the "BankAccount" class is defined with a constructor that takes accountNumber as a parameter and initializes the balance property. The accountNumber property is defined as a val to make it read-only, while the balance property is defined as a var to make it mutable. The "deposit()" method takes an amount parameter, adds it to the balance, and prints the updated balance. The "withdraw()" method takes an amount parameter and checks if the withdrawal amount is less than or equal to the balance. If it is, it subtracts the amount from the balance and prints the updated balance. Otherwise, it prints "Insufficient balance." The "BankAccountApp" object contains the "main()" method where you can test functionality. A BankAccount instance is created, the initial account number and balance are printed, and then deposits and withdrawals are performed.