2. JavaScript: Account Transactions

Create a solution to maintain a bank account's balance.

Implement the *Account* class that manages an account's balance. The class has the following constructor and methods:

1. The constructor *Account(balance)* where parameter *balance* denotes the initial balance of the account.
2. The method *debit(amount)* debits the *amount* from the account and returns true. If insufficient balance, do not debit and return false.
3. The method *getBalance()* returns the current balance.
4. The method *credit(amount)* credits the *amount* to the account.

NOTE: You may assume that the *amount* parameter passed is always positive or 0.

The locked stub code validates the correctness of the *Account* class implementation by performing the following operations:

- *Debit amount*: This operation debits the amount. If the return value is false, it prints 'Insufficient balance'. Otherwise it prints '*<amount>* debited'.
- *Credit amount*: This operation credits the amount and prints '*<amount>* credited'.
- *GetBalance*: This operation gets the current balance and prints 'Current balance is <balance>'.

▼ Input Format For Custom Testing

The first line contains an integer that denotes the initial balance when the *Account* object is created.
The second line contains an integer, *n*, the number of operations to be performed.
Each line *i* of the *n* subsequent lines (where $0 \leq i < n$) contains one of the three operations listed above and their parameters, if any.

▼ Sample Case 0

Sample Input For Custom Testing

```
STDIN            Function
-----            --------
10000       →    initial balance = 10000
3           →    operations count n = 3
Debit 1000  →    first operation = Debit, amount = 1000
Credit 1000
GetBalance
```

Sample Output

```
Account created with initial balance of 10000
1000 debited
1000 credited
Current balance is 10000
```

Explanation

An account object is created with an initial balance of 10000. The debit operation is performed *debit(1000)* which returns true, followed by a credit operation *credit(1000)*. Finally, *getBalance()* is called.

▶ Sample Case 1