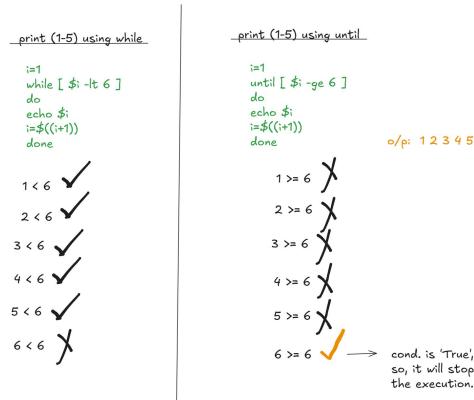


3. until loop:

>> It is similar to while loop.
 >> It does what while loop do, but opposite.
 >> Until loop will execute the block of code as long as the condition is 'False'.
 >> If the condition becomes 'True', it will terminate the loop.



SCRIPT: WRITE A SCRIPT USING UNTIL LOOP TO PRINT EVEN NUMBERS FROM 1-20

```
until [ $i -gt 21 ]
do
if [ $(($i%2)) -eq 0 ]; then
echo $i
fi
i=$((i+1))
done
```

$$\begin{array}{r} 2 \times 20 \\ 20 \\ \hline 0 \end{array} \longrightarrow \%$$

FUNCTIONS

```
details()
read -p "Enter name:" name
read -p "Enter aadhar:" aadhar
read -p "Enter mobile number:" number
read -p "Enter PAN number:" PAN
read -p "Enter address:" address
}
```

BANK → 100 users

Name → read -p "Enter name:" name
Aadhar → read -p "Enter aadhar:" aadhar
mobile →
PAN →
Address →
1 user
user3 user4 user5
details() details() details()

>> Functions allows users to group commands/block of code in a single unit & it can be reused throughout the script.

>> 2 parts are involved in working with functions:

- Defining the function (creating a function)
- Calling the function (making use of the function)

i. Defining the function

```
function-name(){
# statements/block of code
}
```

```
function function-name {
# statements/block of code
}
```

>> Give a function-name and follow it with a parenthesis ().
 >> This parenthesis (), will treat it as a function;

>> Make use of a reserved keyword called as 'function'.
 >> Parenthesis () is nor required in this syntax.

```
eg:
add(){
num1=10
num2=4
echo $((num1+num2))
}
```

add → Function call
./script.sh

```
function bhavana {
num1=10
num2=4
echo $((num1+num2))
}
```

bhavana → Function call

ii. Calling the function

>> Making use of the function.
 >> Calling the function is done by the function's name & it is called inside the script.

>> Function block will invoke & execute, ONLY if the function call is there.

SCRIPT: TAKE 2 NUMBERS FROM USER INPUT & CREATE ADD, SUBSTRACT, MULTIPLY & DIVISION FUNCTION AND PERFORM ALL THESE TASKS.

```
read -p "Enter num1:" num1
read -p "Enter num2:" num2

function add {
echo "Addition of $num1 and $num2 is ${((num1+num2))}"
}

function subtraction(){
echo "Diff of $num1 and $num2 is ${((num1-num2))}"
}
```