* Types of questions from loop (for, while loop)

1. Type-1 → Output - P -
2. Type-2 → Error Correction - P -
3. Type-3 → Conversion (while ⟷ for)
4. Program

* Type-1 : Output →

Text Book (P-38) Q.2.(6)

while loop

```
x = 10
y = 0
while x > y :
    print (x, y)
    x = x - 1
    y = y + 1
```

Sol → x = 10, y = 0, while 10 > 0 → yes → 10 0
x = 9, y = 1
x = 9, y = 1, while 9 > 1 → ✓ → 9 1
x = 8, y = 2 → 8 2
x = 7, y = 3 → 7 3
x = 6, y = 4 → 6 4
x = 5, y = 5 ✗

P-38 Q.2.(g)

for loop

```
for z in range (-500, 500, 100) :
    print (z)
```

Sol → for z → -500, 500, 100
↑ starting value, ↑ UL, ↑ increment

z → -500 → -500
= -400 → -400
= -300 → -300
= -200 → -200

z = -100 → -100
= 0 → 0
= 100 → 100
= 200 → 200
= 300 → 300
= 400 → 400

i. Output →
-500
-400
-300
-200
0
100
200
300
400

Type-2- Error Correction →

Q.1. Re-write the following question after removing syntactical error →

```
for i in ranges ( 1, 10)
    printf (i)
    i i+1
```

↓

Sol→
```
for i in range (1,10):
    print (i)
    # i=i+1 → not required
```

Q.2.
```
for in ranges ( 1, 10, 1)
    print i
    if i = 5
        break
```

Sol→
```
for i in range (1,14,1):
    print (i)
    if i==5:
        break
```

Type-3 - Conversion →

Q.1 Convert / Re-write the following loop into while loop →

```
for i in range (1, 10):
    print (i)
    if i==5:
        break
```

} body of loop

Sol →

```
i = 1
while i < 10:
    print (i)
    if i==5:
        break
    i = i+1
```

Copy ← last line increment

Q.2. Convert / Re-write following loop into for loop →

```
i = 1
while i <= 100:
    if i%2==0:
        i = i*2
    elif i%3==0:
        i = i*3
    print (i)
    i = i+1
```

body of loop

Sol →

```
for i in range (1, 101, 1):
    if i%2==0:
        i = i*2
    elif i%3==0:
        i = i*3
```