```
\qecho 'Question 1'

CREATE OR REPLACE FUNCTION maximumOfNums(num1 integer, num2 integer) RETURNS
integer AS
$$
DECLARE
n1 integer := num1;
n2 integer := num2;
BEGIN
IF n1 > n2 THEN
RETURN n1;
END IF;
IF n2 > n1 THEN
RETURN n2;
END IF;
IF n1 = n2 THEN
RETURN n1;
END IF;
END;
$$ LANGUAGE plpgsql;


SELECT maximumOfNums(42, 28);
SELECT maximumOfNums(33, 55);
SELECT maximumOfNums(33, 33);

\qecho 'Question 2'

CREATE TABLE Customers(id integer, name text, age integer, address text, salary
DECIMAL(6,2));

TABLE Customers;

INSERT INTO Customers VALUES (1, 'Bhargav', 32, 'Ahmedabad', 2000.00),
(2, 'Tarika', 25, 'Delhi', 1500.00),
(3, 'John', 23, 'Kota', 2000.00),
(4, 'Michael', 25, 'Mumbai', 6500.00),
(5, 'Harish', 27, 'Bhopal', 8500.00),
(6, 'Suraj', 22, 'MP', 4500.00);

TABLE Customers;

CREATE OR REPLACE FUNCTION increasedSalary() RETURNS void AS
$$
BEGIN
UPDATE Customers Set salary = salary + 500;
RAISE NOTICE 'UPDATED';
END;
$$ LANGUAGE plpgsql;

-- Test case 1 (incrementing salaries by 500)

SELECT increasedSalary();

TABLE Customers;

-- Test case 2 (incrementing salaries by 500, again)
```

```
SELECT increasedSalary();

TABLE Customers;

\qecho 'Question 3'

CREATE OR REPLACE FUNCTION reversing(num integer) RETURNS integer AS
$$
DECLARE
output integer := 0;
BEGIN
while num > 0 LOOP
output = output * 10 + num % 10;
num = num/10;
END LOOP;
RETURN output;
END;
$$ LANGUAGE plpgsql;

SELECT reversing(123);

SELECT reversing(456);


\qecho 'Question 4'

CREATE OR REPLACE FUNCTION pyramid(num integer) RETURNS void AS
$$
DECLARE
n integer := num;
i integer;
j integer;
output varchar := '*';
BEGIN
FOR i in 1..n LOOP
FOR j in 1..i LOOP
RAISE NOTICE '%', output;
END LOOP;
RAISE NOTICE ' ';
END LOOP;
END;
$$ LANGUAGE plpgsql;

select pyramid(3);
select pyramid(4);



\qecho 'Question 5'

CREATE OR REPLACE FUNCTION find_gcd(num1 integer, num2 integer) RETURNS integer AS
$$
BEGIN
IF num2 = 0 THEN
RETURN num1;
ELSE
RETURN find_gcd(num2, num1 % num2);
END IF;
END;
```

```
$$ LANGUAGE plpgsql;

SELECT find_gcd(8,48);
SELECT find_gcd(144, 12);
```