MySQL will display the source code for your function if you use **show create function** with the function name.

Demo 01: Showing the code for the function `newsalary_6.` Using \G as a command terminator makes this more readable.

```
Show  create function a_emp.newsalary_6 \G
*************************** 1. row ***************************
            Function: newsalary_6
            sql_mode: STRICT_TRANS_TABLES,NO_AUTO_CREATE_USER,NO_ENGINE_SUBSTITUTION
     Create Function: CREATE DEFINER=`a_rose`@`localhost` FUNCTION `newsalary_6`(
   in_salary     decimal (9,2)
 , in_dept      int
 , in_hire_date date ) RETURNS decimal(10,2)
begin
    declare  v_year_hired  decimal (4,0);
    declare  v_new_salary  decimal (10,2);

    set v_year_hired  := extract(year from in_hire_date);
    set v_new_salary := a_emp.newsalary_5(in_salary, in_dept);
    if v_year_hired = extract(year from curdate())
    then
        set v_new_salary := in_salary;
    end if;
    return  v_new_salary;
end
character_set_client: latin1
collation_connection: latin1_swedish_ci
  Database Collation: latin1_swedish_ci
```

Your source code is stored in a table that MySQL maintains and you can access it via the information_schema.routines view.

Demo 02: It is always a good idea- although a bit overwhelming - to describe these views

```
delimiter ;
desc information_schema.routines;
+-------------------------+---------------+------+-----+---------------------+-------+
| Field                   | Type          | Null | Key | Default             | Extra |
+-------------------------+---------------+------+-----+---------------------+-------+
| SPECIFIC_NAME           | varchar(64)   | NO   |     |                     |       |
| ROUTINE_CATALOG         | varchar(512)  | NO   |     |                     |       |
| ROUTINE_SCHEMA          | varchar(64)   | NO   |     |                     |       |
| ROUTINE_NAME            | varchar(64)   | NO   |     |                     |       |
| ROUTINE_TYPE            | varchar(9)    | NO   |     |                     |       |
| DATA_TYPE               | varchar(64)   | NO   |     |                     |       |
| CHARACTER_MAXIMUM_LENGTH| int(21)       | YES  |     | NULL                |       |
| CHARACTER_OCTET_LENGTH  | int(21)       | YES  |     | NULL                |       |
| NUMERIC_PRECISION       | int(21)       | YES  |     | NULL                |       |
| NUMERIC_SCALE           | int(21)       | YES  |     | NULL                |       |
| CHARACTER_SET_NAME      | varchar(64)   | YES  |     | NULL                |       |
| COLLATION_NAME          | varchar(64)   | YES  |     | NULL                |       |
| DTD_IDENTIFIER          | longtext      | YES  |     | NULL                |       |
| ROUTINE_BODY            | varchar(8)    | NO   |     |                     |       |
| ROUTINE_DEFINITION      | longtext      | YES  |     | NULL                |       |
| EXTERNAL_NAME           | varchar(64)   | YES  |     | NULL                |       |
| EXTERNAL_LANGUAGE       | varchar(64)   | YES  |     | NULL                |       |
| PARAMETER_STYLE         | varchar(8)    | NO   |     |                     |       |
| IS_DETERMINISTIC        | varchar(3)    | NO   |     |                     |       |
| SQL_DATA_ACCESS         | varchar(64)   | NO   |     |                     |       |
| SQL_PATH                | varchar(64)   | YES  |     | NULL                |       |
| SECURITY_TYPE           | varchar(7)    | NO   |     |                     |       |
| CREATED                 | datetime      | NO   |     | 0000-00-00 00:00:00 |       |
| LAST_ALTERED            | datetime      | NO   |     | 0000-00-00 00:00:00 |       |
| SQL_MODE                | varchar(8192) | NO   |     |                     |       |
```

```
| ROUTINE_COMMENT          | longtext      | NO   |     | NULL                |       |
| DEFINER                  | varchar(77)   | NO   |     |                     |       |
| CHARACTER_SET_CLIENT     | varchar(32)   | NO   |     |                     |       |
| COLLATION_CONNECTION     | varchar(32)   | NO   |     |                     |       |
| DATABASE_COLLATION       | varchar(32)   | NO   |     |                     |       |
+--------------------------+---------------+------+-----+---------------------+-------+
30 rows in set (0.00 sec)
```

Demo 03: Let's look at what we get for one of our functions; I have highlighted some of the easier columns to understand. We could filter for functions; functions that return a certain data type; functions in a specified schema

```
select *
from information_schema.routines
where routine_name = 'newsalary_6'\G
*************************** 1. row ***************************
           SPECIFIC_NAME: newsalary_6
         ROUTINE_CATALOG: def
          ROUTINE_SCHEMA: a_emp
            ROUTINE_NAME: newsalary_6
            ROUTINE_TYPE: FUNCTION
               DATA_TYPE: decimal
CHARACTER_MAXIMUM_LENGTH: NULL
  CHARACTER_OCTET_LENGTH: NULL
       NUMERIC_PRECISION: 10
           NUMERIC_SCALE: 2
      CHARACTER_SET_NAME: NULL
          COLLATION_NAME: NULL
          DTD_IDENTIFIER: decimal(10,2)
            ROUTINE_BODY: SQL
      ROUTINE_DEFINITION: begin
    declare  v_year_hired  decimal (4,0);
    declare  v_new_salary  decimal (10,2);

    set v_year_hired  := extract(year from in_hire_date);
    set v_new_salary := a_emp.newsalary_5(in_salary, in_dept);
    if v_year_hired = extract(year from curdate())
    then
        set v_new_salary := in_salary;
    end if;
    return  v_new_salary;
end
           EXTERNAL_NAME: NULL
       EXTERNAL_LANGUAGE: NULL
         PARAMETER_STYLE: SQL
        IS_DETERMINISTIC: NO
         SQL_DATA_ACCESS: CONTAINS SQL
                SQL_PATH: NULL
           SECURITY_TYPE: DEFINER
                 CREATED: 2013-06-09 21:19:16
            LAST_ALTERED: 2013-06-09 21:19:16
                SQL_MODE:
STRICT_TRANS_TABLES,NO_AUTO_CREATE_USER,NO_ENGINE_SUBSTITUTION
         ROUTINE_COMMENT:
                 DEFINER: a_rose@localhost
    CHARACTER_SET_CLIENT: latin1
    COLLATION_CONNECTION: latin1_swedish_ci
      DATABASE_COLLATION: latin1_swedish_ci
1 row in set (0.03 sec)
```

Demo 04: This displays the body of a function but not the header. (I have narrowed the header columns. The routine code body is displayed in a single cell,

```
Select routine_definition
From information_schema.routines
Where routine_type = 'function'
and  routine_name = 'newsalary_6'
and routine_schema = 'a_emp';
+------------------------------------------------------------------------------+
| routine_definition                                                           |
+------------------------------------------------------------------------------+
| begin
    declare  v_year_hired  decimal (4,0);
    declare  v_new_salary  decimal (10,2);

    set v_year_hired  := extract(year from in_hire_date);
    set v_new_salary := a_emp.newsalary_5(in_salary, in_dept);
    if v_year_hired = extract(year from curdate())
    then
        set v_new_salary := in_salary;
    end if;
    return  v_new_salary;
end |
+------------------------------------------------------------------------------+
```

Demo 05: Parameter information

```
desc information_schema.parameters;
+-------------------------+--------------+------+-----+---------+-------+
| Field                   | Type         | Null | Key | Default | Extra |
+-------------------------+--------------+------+-----+---------+-------+
| SPECIFIC_CATALOG        | varchar(512) | NO   |     |         |       |
| SPECIFIC_SCHEMA         | varchar(64)  | NO   |     |         |       |
| SPECIFIC_NAME           | varchar(64)  | NO   |     |         |       |
| ORDINAL_POSITION        | int(21)      | NO   |     | 0       |       |
| PARAMETER_MODE          | varchar(5)   | YES  |     | NULL    |       |
| PARAMETER_NAME          | varchar(64)  | YES  |     | NULL    |       |
| DATA_TYPE               | varchar(64)  | NO   |     |         |       |
| CHARACTER_MAXIMUM_LENGTH| int(21)      | YES  |     | NULL    |       |
| CHARACTER_OCTET_LENGTH  | int(21)      | YES  |     | NULL    |       |
| NUMERIC_PRECISION       | int(21)      | YES  |     | NULL    |       |
| NUMERIC_SCALE           | int(21)      | YES  |     | NULL    |       |
| CHARACTER_SET_NAME      | varchar(64)  | YES  |     | NULL    |       |
| COLLATION_NAME          | varchar(64)  | YES  |     | NULL    |       |
| DTD_IDENTIFIER          | longtext     | NO   |     | NULL    |       |
| ROUTINE_TYPE            | varchar(9)   | NO   |     |         |       |
+-------------------------+--------------+------+-----+---------+-------+
```

Demo 06: This shows you information about the parameters for this function..

```
select
  parameter_name
, ordinal_position
, data_type
, parameter_mode
from information_schema.parameters
where specific_name = 'newsalary_6'
and specific_schema = 'a_emp';
```

```
+----------------+------------------+-----------+----------------+
| parameter_name | ordinal_position | data_type | parameter_mode |
+----------------+------------------+-----------+----------------+
| NULL           |                0 | decimal   | NULL           |
| in_salary      |                1 | decimal   | IN             |
| in_dept        |                2 | int       | IN             |
| in_hire_date   |                3 | date      | IN             |
+----------------+------------------+-----------+----------------+
```

Demo 07: Looking for routines that use certain data types

```
Select
  specific_schema, specific_name
, parameter_name
, ordinal_position
, parameter_mode
From information_schema.parameters
Where data_type = 'date'
;
SELECT ROWS
+-----------------+---------------+----------------+------------------+----------------+
| specific_schema | specific_name | parameter_name | ordinal_position | parameter_mode |
+-----------------+---------------+----------------+------------------+----------------+
| a_emp           | FutureDate2   | NULL           |                0 | NULL           |
| a_emp           | FutureDate2   | p_date         |                1 | IN             |
| a_emp           | newsalary_6   | in_hire_date   |                3 | IN             |
| a_testbed       | FutureDate    | NULL           |                0 | NULL           |
| a_testbed       | FutureDate2   | NULL           |                0 | NULL           |
+-----------------+---------------+----------------+------------------+----------------+

Select
  specific_schema, specific_name
, parameter_name
, ordinal_position
, parameter_mode
From information_schema.parameters
Where data_type = 'int'
And specific_schema = 'a_emp';
+-----------------+-------------------+----------------+------------------+----------------+
| specific_schema | specific_name     | parameter_name | ordinal_position | parameter_mode |
+-----------------+-------------------+----------------+------------------+----------------+
| a_emp           | BookSize          | pagecount_in   |                1 | IN             |
| a_emp           | DeptEmployeeCount | NULL           |                0 | NULL           |
| a_emp           | DeptEmployeeCount | p_dept_id      |                1 | IN             |
| a_emp           | empjobtitle       | in_emp_id      |                1 | IN             |
| a_emp           | empjobtitle_v2    | in_emp_id      |                1 | IN             |
| a_emp           | FutureDate2       | p_y            |                2 | IN             |
| a_emp           | FutureDate2       | p_m            |                3 | IN             |
| a_emp           | FutureDate2       | p_d            |                4 | IN             |
| a_emp           | newsalary_5       | in_dept        |                2 | IN             |
| a_emp           | newsalary_5_V2    | in_dept        |                2 | IN             |
| a_emp           | newsalary_6       | in_dept        |                2 | IN             |
+-----------------+-------------------+----------------+------------------+----------------+
```