```
/* 1. Create Database schema called ClassAssignment */

CREATE DATABASE IF NOT EXISTS ClassAssignment;


/* 2. Create a table called Project with the following columns:
        project_num INT NOT NULL PRIMARY KEY
        project_code CHAR(4)
        project_title VARCHAR(45)
        first_name VARCHAR(45)
        last_name VARCHAR(45)
        project_budget DECIMAL(5, 2) */

CREATE TABLE IF NOT EXISTS Project
(
   project_num INT NOT NULL PRIMARY KEY,
   project_code CHAR(4) ,
   project_title VARCHAR(45),
        first_name VARCHAR(45),
        last_name VARCHAR(45),
        project_budget DECIMAL(5, 2)
);

-- DESC Project;


/* 3. Modify project_num to auto_increment and also auto_increment starts from 10. */

ALTER TABLE Project
MODIFY project_num INT AUTO_INCREMENT;

ALTER TABLE Project
AUTO_INCREMENT = 10;

-- DESC Project;


/* 4. Modify project_budget datatype from decimal (5, 2) to (10, 2). */

ALTER TABLE Project
MODIFY project_budget DEC(10, 2);

-- DESC Project;


/* 5. Insert following values into the Project table.
DO NOT insert project_num. Auto_increment should start from 10 */
```

```
INSERT INTO Project (project_code, project_title, first_name, last_name, project_budget)
VALUES ('PC01', 'DIA', 'John', 'Smith', 10000.99);

INSERT INTO Project (project_code, project_title, first_name, last_name, project_budget)
VALUES ('PC02', 'CHF', 'Tim', 'Cook', 12000.50);

INSERT INTO Project (project_code, project_title, first_name, last_name, project_budget)
VALUES ('PC03', 'AST', 'Rhonda', 'Smith', 8000.40);

-- SELECT project_code, project_title, first_name, last_name, project_budget
FROM Project;


/* 6. Create a table PayRoll with the following info:
        employee_num INT PRIMARY KEY AUTO_INCREMENT
        job_id INT NOT NULL
        job_desc VARCHAR(40)
        emp_pay DECIMAL (10, 2) */

CREATE TABLE IF NOT EXISTS PayRoll
(
        employee_num INT PRIMARY KEY AUTO_INCREMENT,
        job_id INT NOT NULL,
        job_desc VARCHAR(40),
        emp_pay DECIMAL (10, 2)
);

-- DESC PayRoll;


/* 7. Alter PayRoll table with the following, make sure to write each script separately.
        i. Add constraint on emp_pay so that only value greater than 10,000 can be inserted.
        ii. Add constraint on job_desc so that default value set to 'Data Analyst'.
        iii. Add column pay_date (DATE) after job_desc. */

ALTER TABLE PayRoll
ADD CONSTRAINT emp_pay_chk CHECK (emp_pay > 10000);

ALTER TABLE PayRoll
        ALTER job_desc SET DEFAULT 'Data Analyst';

ALTER TABLE PayRoll
        ADD COLUMN pay_date DATE AFTER job_desc;


-- DESC PayRoll;
```

/* 8. Add Foreign Key constraint in PayRoll table with job_id column referencing to project_num column in Project table. */

```
ALTER TABLE PayRoll
ADD CONSTRAINT fk_job FOREIGN KEY (job_id) REFERENCES Project (project_num);

-- DESC PayRoll;
```

/* 9. Insert following values into PayRoll table.
DO NOT insert employee_num and job_desc, those should be auto populated using auto_increment and default values, respectively. */

```
INSERT INTO PayRoll (job_id, pay_date, emp_pay)
VALUES (10, '2023-03-05', 12000.99);

INSERT INTO PayRoll (job_id, pay_date, emp_pay)
VALUES (11, '2023-03-05', 14000.99);

INSERT INTO PayRoll (job_id, pay_date, emp_pay)
VALUES (12, '2023-03-05', 16000.99);

-- SELECT *
FROM PayRoll;
```

/* 10. Update emp_pay in PayRoll table for employee_num = 2 with 10% emp_pay increase. */

```
UPDATE PayRoll
  SET emp_pay = (emp_pay * 1.1)
  WHERE employee_num = 2;

-- SELECT *
FROM PayRoll
WHERE employee_num = 2;
```

/* 11. Create Project_backup table from Project table you created above using bulk insert statement only for last_name ‘Smith'. */

```
CREATE TABLE Project_backup
        SELECT *
   FROM Project
   WHERE last_name = 'Smith';

-- DESC Project_backup;
```

/* 12. Create VIEW as PayRoll_View from PayRoll table you created above.
However, your VIEW should only contain job_id, job_desc and pay_date for job_id > 10. */

```
CREATE VIEW PayRoll_View AS
        SELECT job_id, job_desc, pay_date
   FROM PayRoll
   WHERE job_id > 10;

-- SELECT *
FROM PayRoll_View;
```

/* 13. Create Index for pay_date on PayRoll table. */

```
CREATE INDEX IX_pay_date ON PayRoll (pay_date);

-- SHOW INDEX FROM PayRoll;
```

/* 14. Delete all data from project_backup table but keep the table structure. */

```
TRUNCATE TABLE project_backup;

-- SELECT *
FROM project_backup;
```

/* 15. Write a DELETE script to delete a row from Project table where project_num = 10.
If there is an error, give a short explanation of what/why about error msg? */

```
DELETE FROM Project WHERE project_num = 10;

-- SELECT *
FROM Project
WHERE project_num = 10;
```

/* 16. Solve the question 15 above without error, i.e. write a script how you can delete. */

There was no error in question 15!