LONDON
METROPOLITAN
UNIVERSITY

islington college
(इस्लिङ्टन कलेज)

Module Code & Module Title

Level 6 – Data and Web Development

Assessment Type: Milestone 2

Semester: 5th Sem

2024/25 Autumn

Student Name: Avinav Baral

London Met ID: 22085508

College ID: np01cp4s230116@isligtoncollege.edu.np

Assignment Due Date: Friday, January 3, 2025

Assignment Submission Date: Friday, January 3, 2025

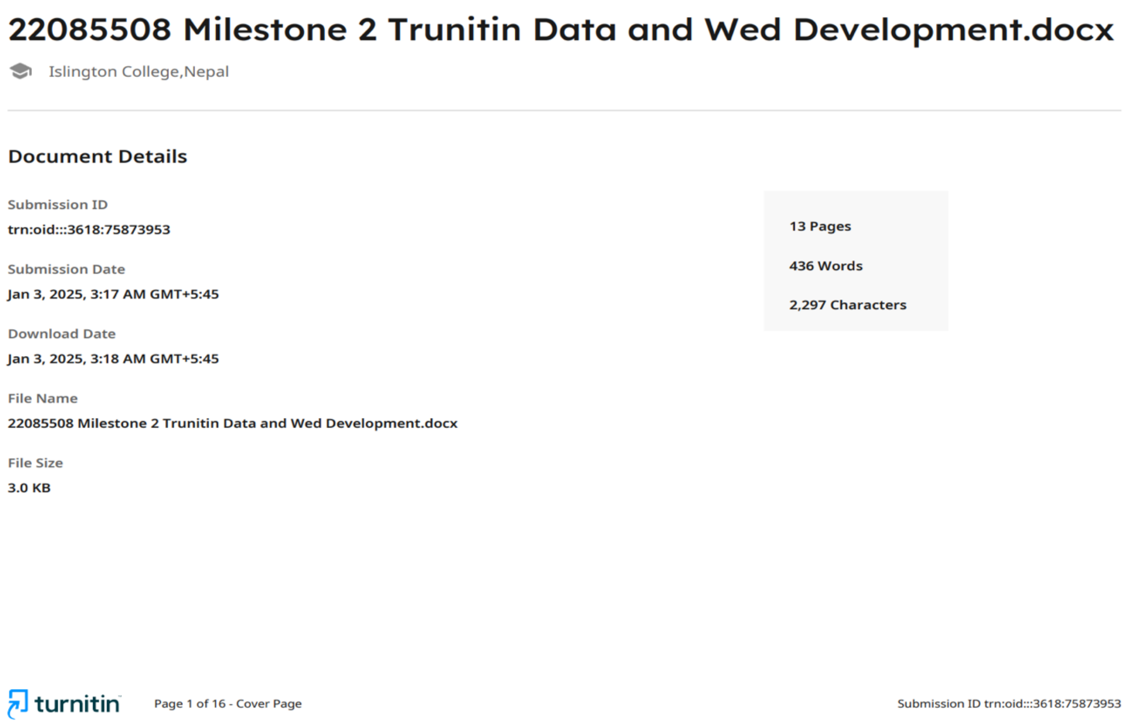
Submitted To: Lekhnath Katuwal

Word Count (Where Required): 498

I confirm that I understand my coursework needs to be submitted online via MST Classroom under the relevant module page before the deadline in order for my assignment to be accepted and marked. I am fully aware that late submissions will be treated as non-submission and a mark of zero will be awarded.

Table of Contents

Table of Figures



22085508 Milestone 2 Trunitin Data and Wed Development.docx

Islington College,Nepal

Document Details

Submission ID
trn:oid:::3618:75873953

Submission Date
Jan 3, 2025, 3:17 AM GMT+5:45

Download Date
Jan 3, 2025, 3:18 AM GMT+5:45

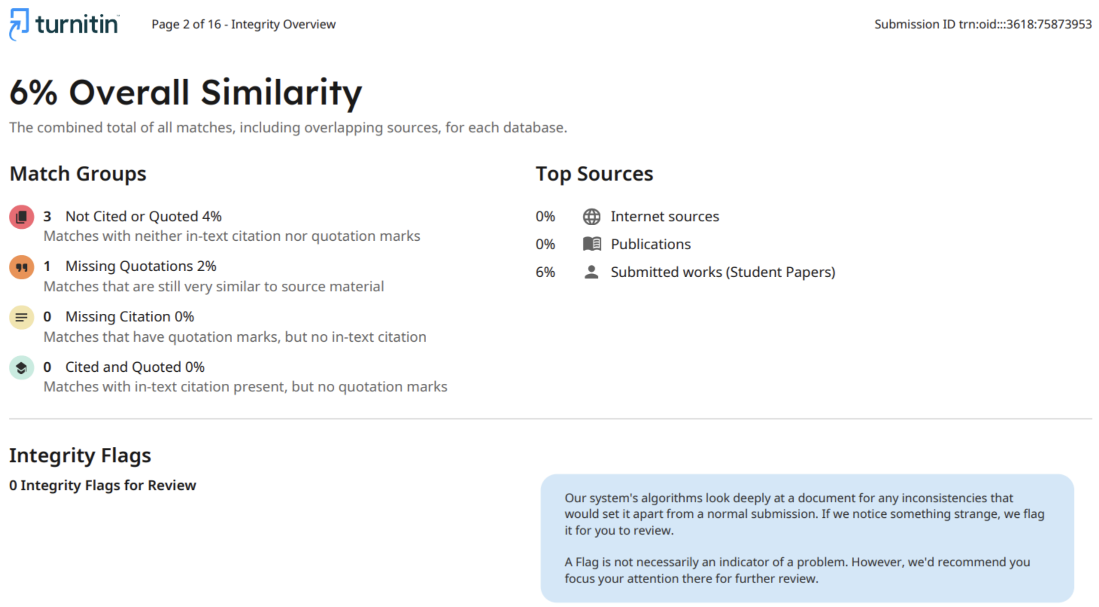
File Name
22085508 Milestone 2 Trunitin Data and Wed Development.docx

File Size
3.0 KB

13 Pages

436 Words

2,297 Characters



6% Overall Similarity

The combined total of all matches, including overlapping sources, for each database.

Match Groups

- 3 Not Cited or Quoted 4%
  Matches with neither in-text citation nor quotation marks
- 1 Missing Quotations 2%
  Matches that are still very similar to source material
- 0 Missing Citation 0%
  Matches that have quotation marks, but no in-text citation
- 0 Cited and Quoted 0%
  Matches with in-text citation present, but no quotation marks

Top Sources

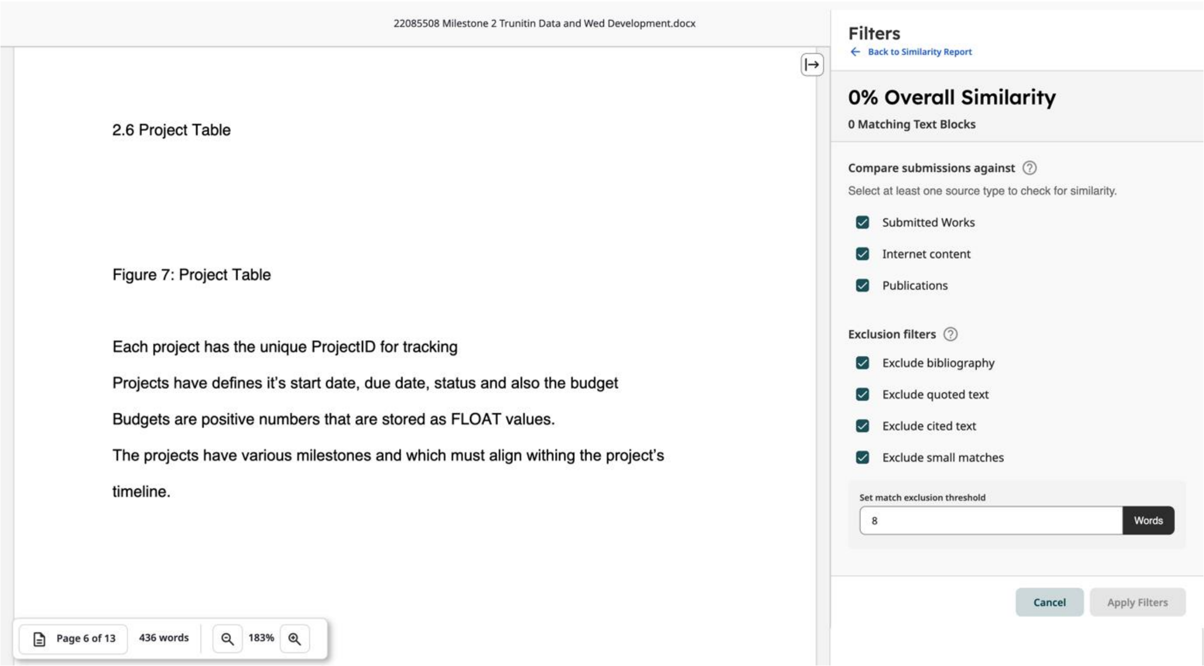
0%	Internet sources
0%	Publications
6%	Submitted works (Student Papers)

Integrity Flags

0 Integrity Flags for Review

Our system's algorithms look deeply at a document for any inconsistencies that would set it apart from a normal submission. If we notice something strange, we flag it for you to review.

A Flag is not necessarily an indicator of a problem. However, we'd recommend you focus your attention there for further review.



22085508 Milestone 2 Trunitin Data and Wed Development.docx

2.6 Project Table

Figure 7: Project Table

Each project has the unique ProjectID for tracking

Projects have defines it's start date, due date, status and also the budget

Budgets are positive numbers that are stored as FLOAT values.

The projects have various milestones and which must align withing the project's timeline.

Page 6 of 13 436 words 183%

Filters

← Back to Similarity Report

0% Overall Similarity

0 Matching Text Blocks

Compare submissions against

Select at least one source type to check for similarity.

☑ Submitted Works

☑ Internet content

☑ Publications

Exclusion filters

☑ Exclude bibliography

☑ Exclude quoted text

☑ Exclude cited text

☑ Exclude small matches

Set match exclusion threshold

8 Words

1. Final ERD

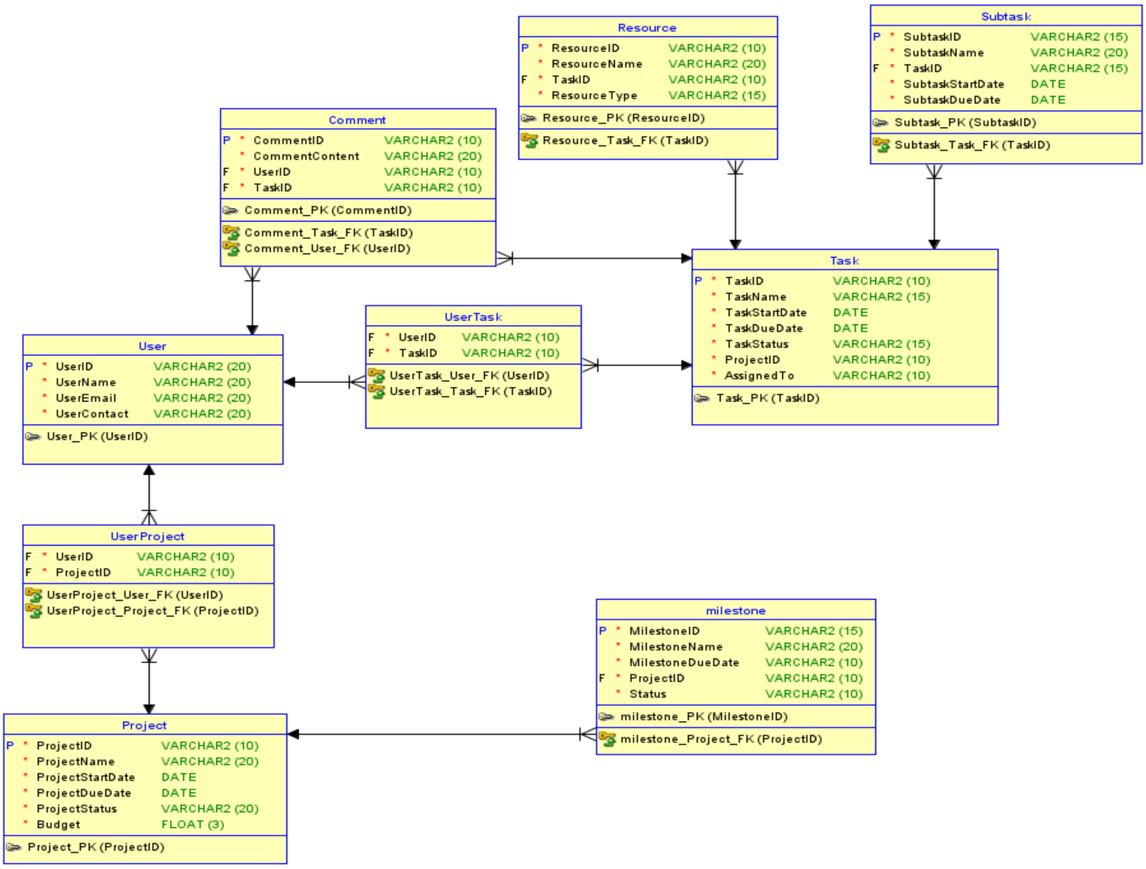

Resource
P * ResourceID VARCHAR2 (10)
* ResourceName VARCHAR2 (20)
F * TaskID VARCHAR2 (10)
* ResourceType VARCHAR2 (15)
Resource_PK (ResourceID)
Resource_Task_FK (TaskID)
Subtask
P * SubtaskID VARCHAR2 (15)
* SubtaskName VARCHAR2 (20)
F * TaskID VARCHAR2 (15)
* SubtaskStartDate DATE
* SubtaskDueDate DATE
Subtask_PK (SubtaskID)
Subtask_Task_FK (TaskID)
Comment
P * CommentID VARCHAR2 (10)
* CommentContent VARCHAR2 (20)
F * UserID VARCHAR2 (10)
F * TaskID VARCHAR2 (10)
Comment_PK (CommentID)
Comment_Task_FK (TaskID)
Comment_User_FK (UserID)
Task
P * TaskID VARCHAR2 (10)
* TaskName VARCHAR2 (15)
* TaskStartDate DATE
* TaskDueDate DATE
* TaskStatus VARCHAR2 (15)
* ProjectID VARCHAR2 (10)
* AssignedTo VARCHAR2 (10)
Task_PK (TaskID)
UserTask
F * UserID VARCHAR2 (10)
F * TaskID VARCHAR2 (10)
UserTask_User_FK (UserID)
UserTask_Task_FK (TaskID)
User
P * UserID VARCHAR2 (20)
* UserName VARCHAR2 (20)
* UserEmail VARCHAR2 (20)
* UserContact VARCHAR2 (20)
User_PK (UserID)
UserProject
F * UserID VARCHAR2 (10)
F * ProjectID VARCHAR2 (10)
UserProject_User_FK (UserID)
UserProject_Project_FK (ProjectID)
milestone
P * MilestoneID VARCHAR2 (15)
* MilestoneName VARCHAR2 (20)
* MilestoneDueDate VARCHAR2 (10)
F * ProjectID VARCHAR2 (10)
* Status VARCHAR2 (10)
milestone_PK (MilestoneID)
milestone_Project_FK (ProjectID)
Project
P * ProjectID VARCHAR2 (10)
* ProjectName VARCHAR2 (20)
* ProjectStartDate DATE
* ProjectDueDate DATE
* ProjectStatus VARCHAR2 (20)
* Budget FLOAT (3)
Project_PK (ProjectID)

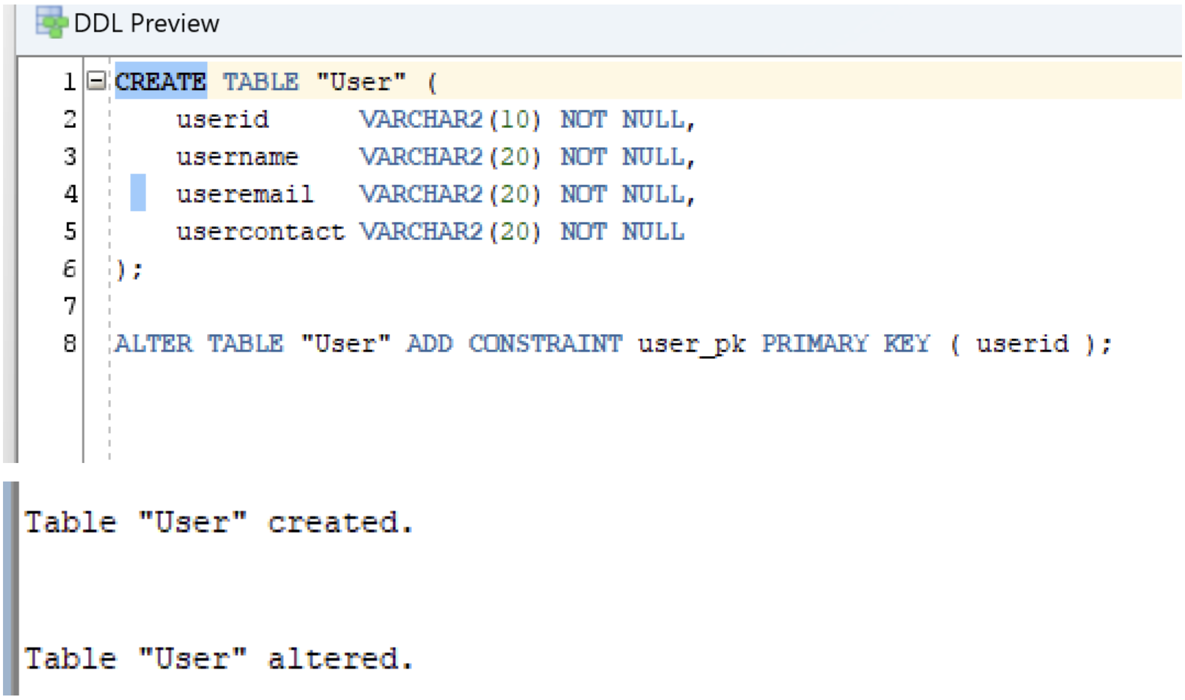
Figure 1: Final ERD

Assumptions

- Each of those tables have its own primary key to ensure the uniqueness and to prevent duplicate records
- It contains the foreign key and these foreign key maintains the relationships between tables to ensure referential integrity.
- All attributes are defined with their appropriate data types also the sizes example VARCHAR (10)
- This also supports adding the more data.
- Considering the many to many relation between the user and task and user and project, here the UserTask is one of the bridge entity and UserProject is another bridge entity between them.

2. DDL Preview

2.1 User table



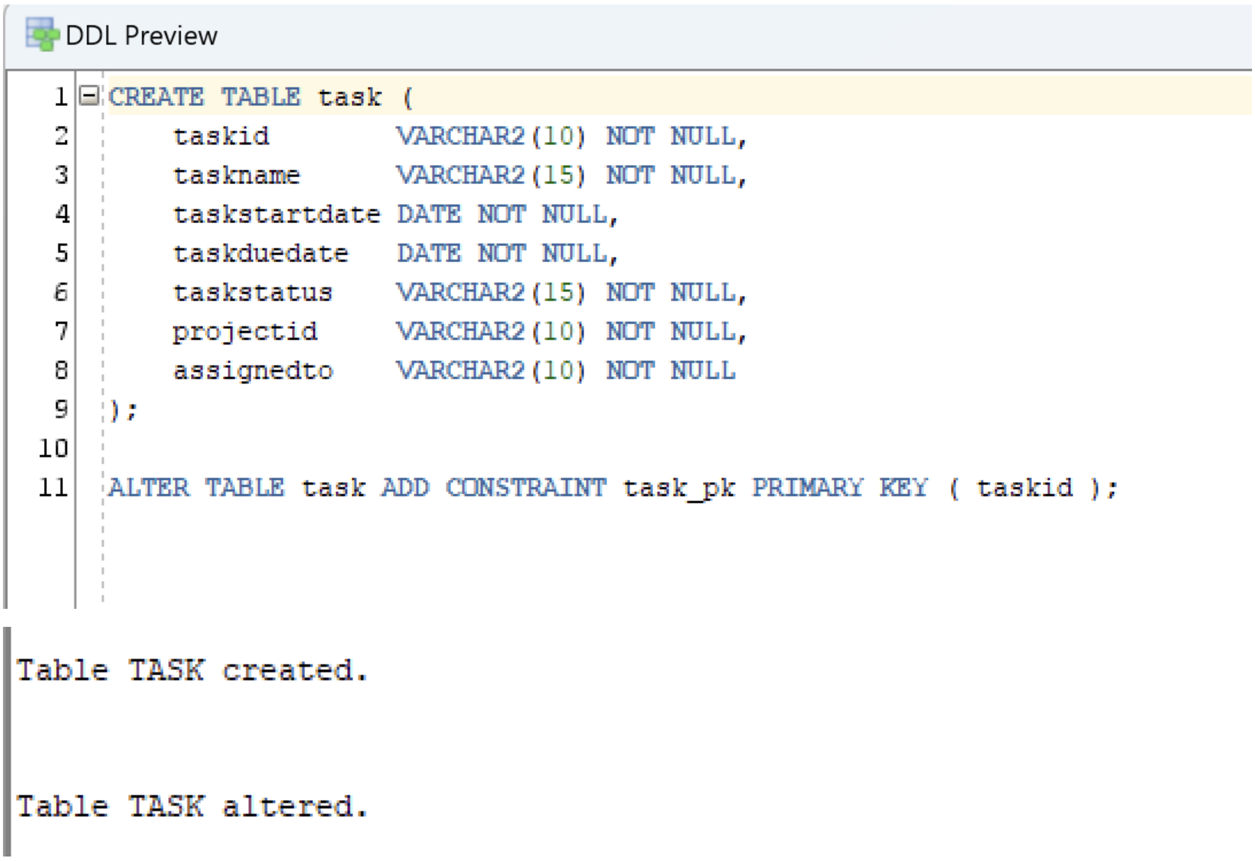
```
CREATE TABLE "User" (
    userid      VARCHAR2(10) NOT NULL,
    username    VARCHAR2(20) NOT NULL,
    useremail   VARCHAR2(20) NOT NULL,
    usercontact VARCHAR2(20) NOT NULL
);

ALTER TABLE "User" ADD CONSTRAINT user_pk PRIMARY KEY ( userid );
```

```
Table "User" created.


Table "User" altered.
```

Figure 2: User table

- Each user has a unique “**Userid**” for its own identification
- Users must provide the valid email address and contact numbers
- Users can also participate in multiple projects using the linking tables

2.2 Task Table



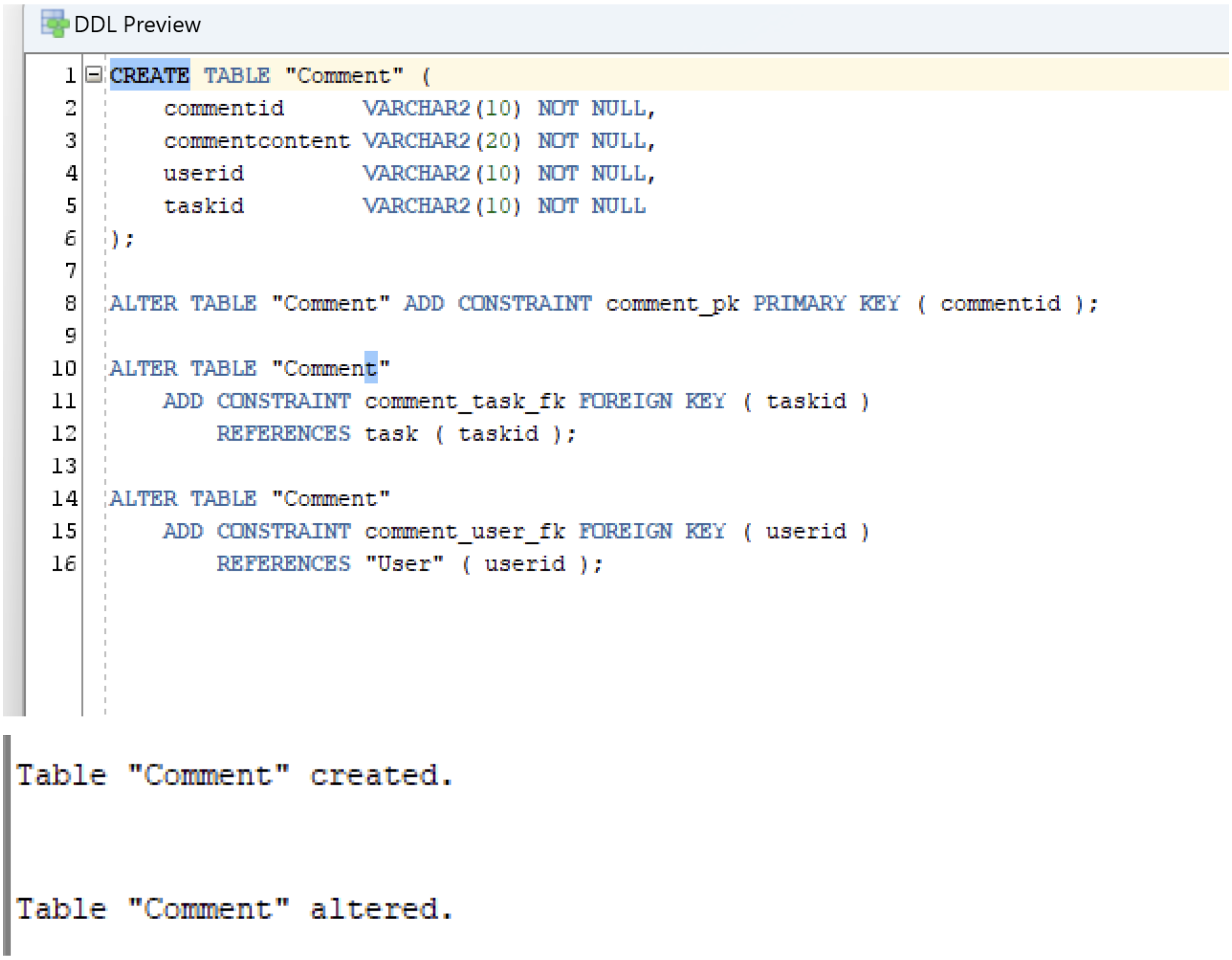
```
CREATE TABLE task (
    taskid        VARCHAR2(10) NOT NULL,
    taskname      VARCHAR2(15) NOT NULL,
    taskstartdate DATE NOT NULL,
    taskduedate   DATE NOT NULL,
    taskstatus    VARCHAR2(15) NOT NULL,
    projectid     VARCHAR2(10) NOT NULL,
    assignedto    VARCHAR2(10) NOT NULL
);

ALTER TABLE task ADD CONSTRAINT task_pk PRIMARY KEY ( taskid );
```

```
Table TASK created.


Table TASK altered.
```

Figure 3: Task Table

- Each task has a unique “**taskid**” for its own identification
- Tasks are associated with specific projects through the projectID
- Tasks have start dates, due dates, and statuses such as Pending, In Progress, and Completed.
- Tasks can be assigned to only one user (Assigned to)

2.3 Comment Table



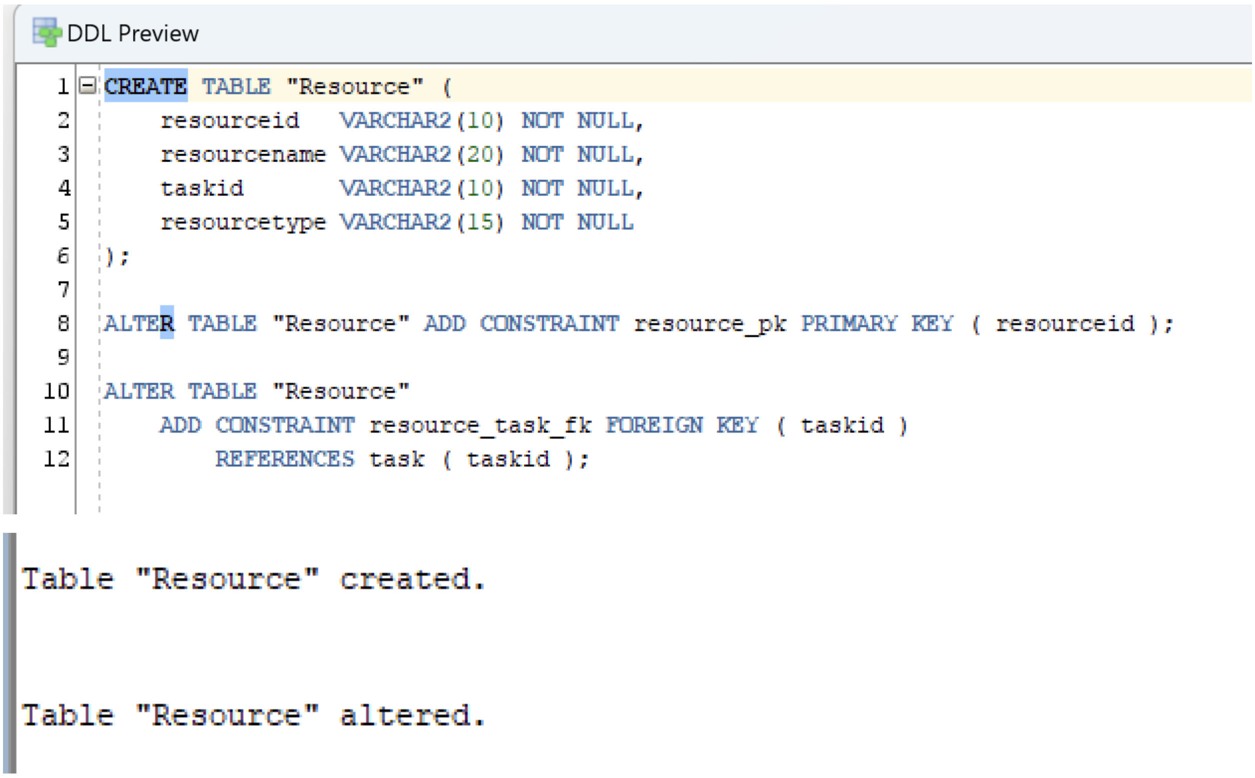
```
CREATE TABLE "Comment" (
    commentid      VARCHAR2(10) NOT NULL,
    commentcontent VARCHAR2(20) NOT NULL,
    userid         VARCHAR2(10) NOT NULL,
    taskid         VARCHAR2(10) NOT NULL
);

ALTER TABLE "Comment" ADD CONSTRAINT comment_pk PRIMARY KEY ( commentid );

ALTER TABLE "Comment"
    ADD CONSTRAINT comment_task_fk FOREIGN KEY ( taskid )
        REFERENCES task ( taskid );

ALTER TABLE "Comment"
    ADD CONSTRAINT comment_user_fk FOREIGN KEY ( userid )
        REFERENCES "User" ( userid );
```

```
Table "Comment" created.


Table "Comment" altered.
```

Figure 4: Comment Table

- Each comment has the unique “**CommmentID**”
- These comments are linked with specific tasks and specific users
- Comments contents is limited as it have VARCHAR (20) so, limited text are and also only supports text.

2.4 Resource Table



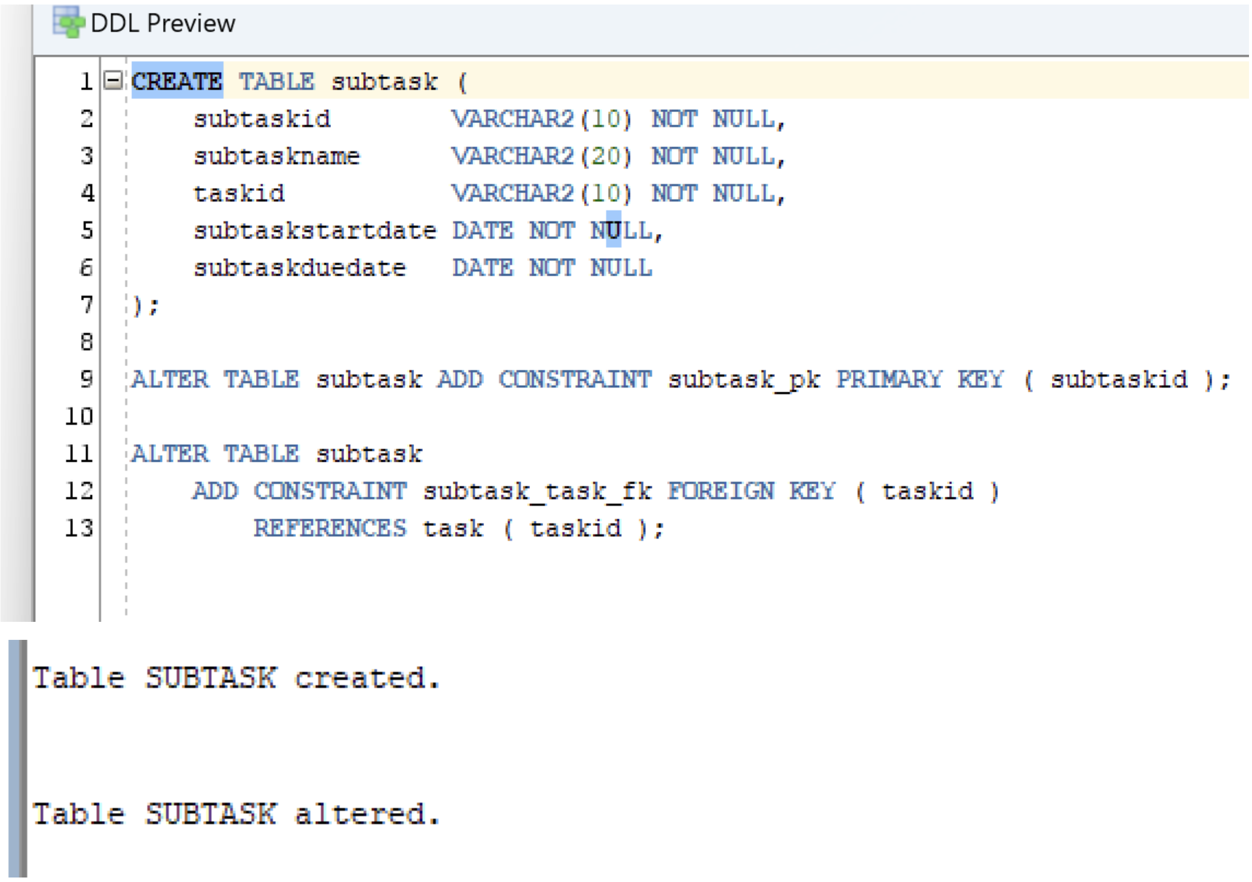
```
CREATE TABLE "Resource" (
    resourceid   VARCHAR2(10) NOT NULL,
    resourcename VARCHAR2(20) NOT NULL,
    taskid       VARCHAR2(10) NOT NULL,
    resourcetype VARCHAR2(15) NOT NULL
);

ALTER TABLE "Resource" ADD CONSTRAINT resource_pk PRIMARY KEY ( resourceid );

ALTER TABLE "Resource"
    ADD CONSTRAINT resource_task_fk FOREIGN KEY ( taskid )
        REFERENCES task ( taskid );
```

```
Table "Resource" created.


Table "Resource" altered.
```

Figure 5: Resource Table

- Resources have the unique “**ResourceID**” for its own identification as the primary key
- Resources can be categorized as humans, or other materials

2.5 Subtask Table



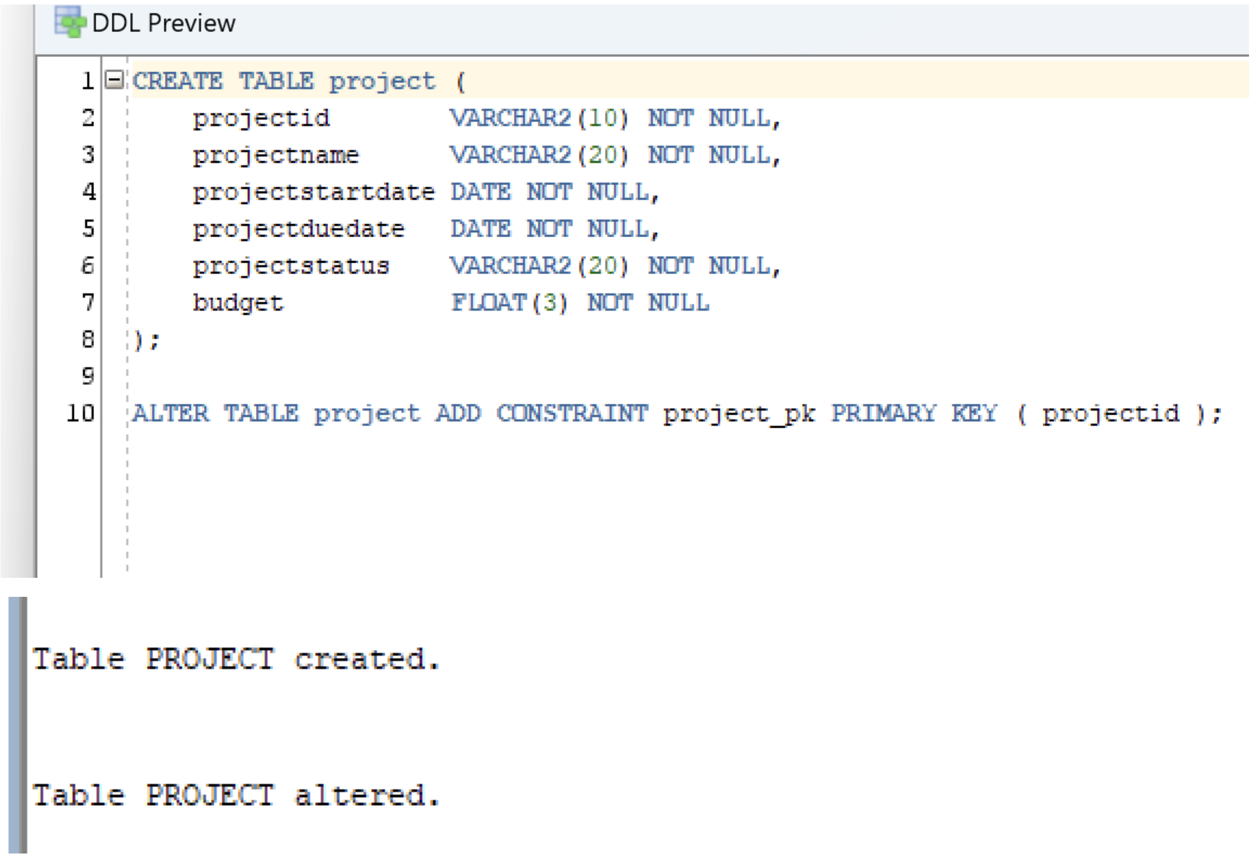
```
CREATE TABLE subtask (
    subtaskid        VARCHAR2(10) NOT NULL,
    subtaskname      VARCHAR2(20) NOT NULL,
    taskid           VARCHAR2(10) NOT NULL,
    subtaskstartdate DATE NOT NULL,
    subtaskduedate   DATE NOT NULL
);

ALTER TABLE subtask ADD CONSTRAINT subtask_pk PRIMARY KEY ( subtaskid );

ALTER TABLE subtask
    ADD CONSTRAINT subtask_task_fk FOREIGN KEY ( taskid )
        REFERENCES task ( taskid );
```

```
Table SUBTASK created.


Table SUBTASK altered.
```

Figure 6: Subtask Table

- Subtask has unique “**SubtaskID**”
- Subtasks are linked to tasks through TaskID.
- Subtasks must have valid start and due dates within the parent task’s timeline.

2.6 Project Table



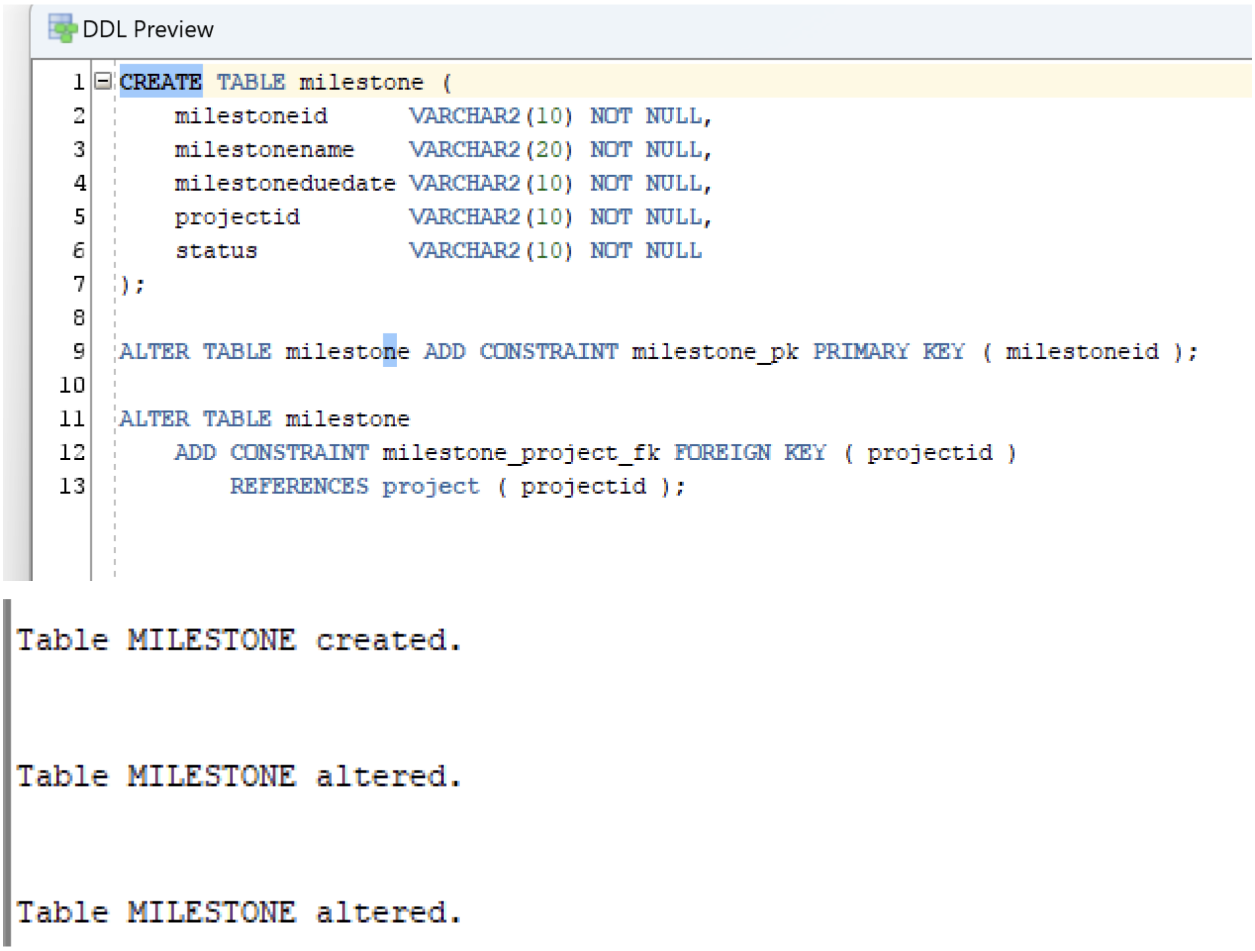
```
CREATE TABLE project (
    projectid        VARCHAR2(10) NOT NULL,
    projectname      VARCHAR2(20) NOT NULL,
    projectstartdate DATE NOT NULL,
    projectduedate   DATE NOT NULL,
    projectstatus    VARCHAR2(20) NOT NULL,
    budget           FLOAT(3) NOT NULL
);

ALTER TABLE project ADD CONSTRAINT project_pk PRIMARY KEY ( projectid );
```

```
Table PROJECT created.


Table PROJECT altered.
```

Figure 7: Project Table

- Each project has the unique ProjectID for tracking
- Projects have defines it’s start date, due date, status and also the budget
- Budgets are positive numbers that are stored as FLOAT values.
- The projects have various milestones and which must align withing the project’s timeline.

2.7 Milestone Table



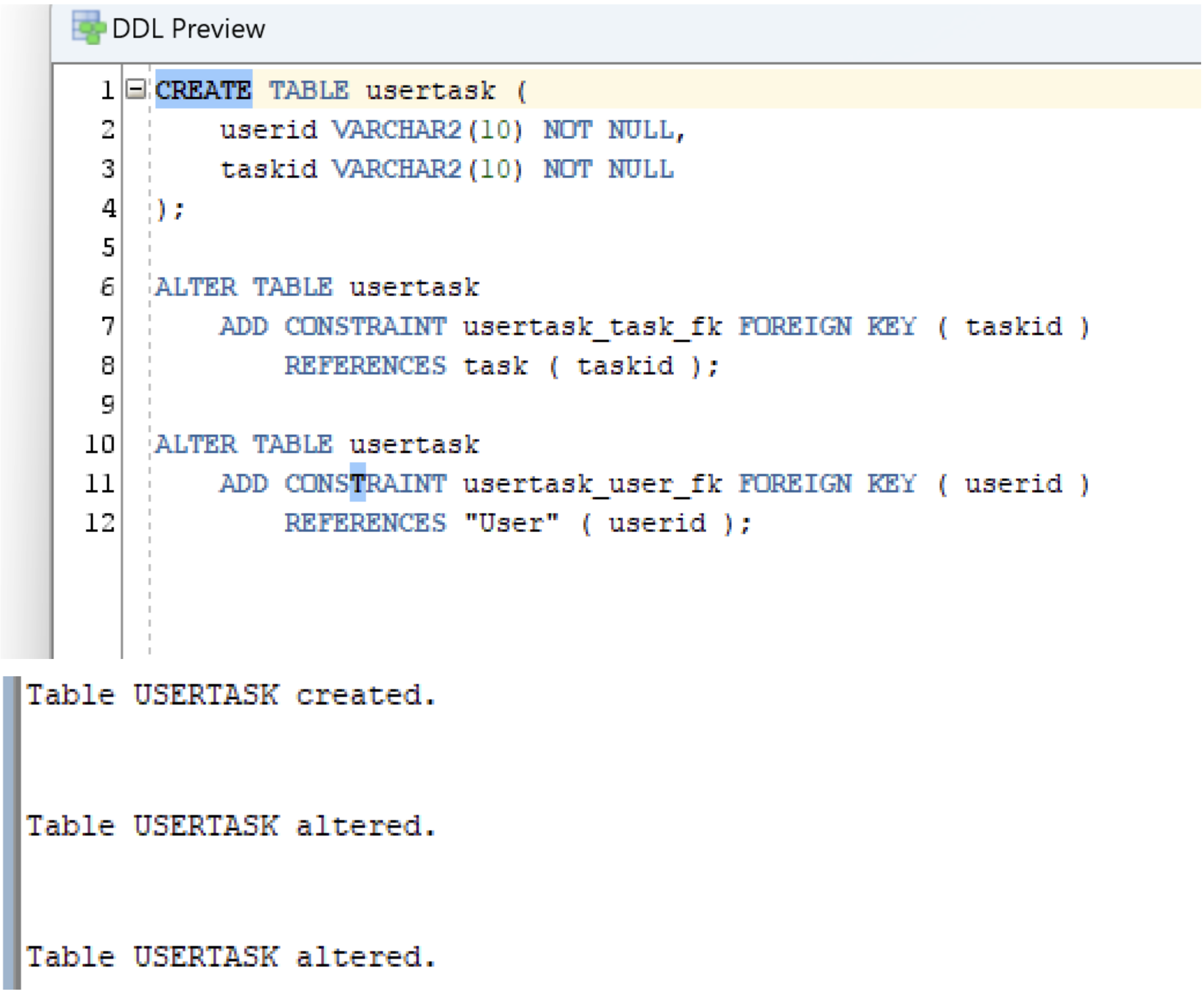
```
CREATE TABLE milestone (
    milestoneid      VARCHAR2(10) NOT NULL,
    milestonename    VARCHAR2(20) NOT NULL,
    milestoneduedate VARCHAR2(10) NOT NULL,
    projectid        VARCHAR2(10) NOT NULL,
    status           VARCHAR2(10) NOT NULL
);

ALTER TABLE milestone ADD CONSTRAINT milestone_pk PRIMARY KEY ( milestoneid );

ALTER TABLE milestone
    ADD CONSTRAINT milestone_project_fk FOREIGN KEY ( projectid )
        REFERENCES project ( projectid );
```

```
Table MILESTONE created.


Table MILESTONE altered.


Table MILESTONE altered.
```

Figure 8: Milestone Table

- Each of these milestone have a unique “**MilestoneID**”
- Milestones are associated with only one project
- These milestones due dates cannot exceed the project’s end date.

2.8 UserTask Table

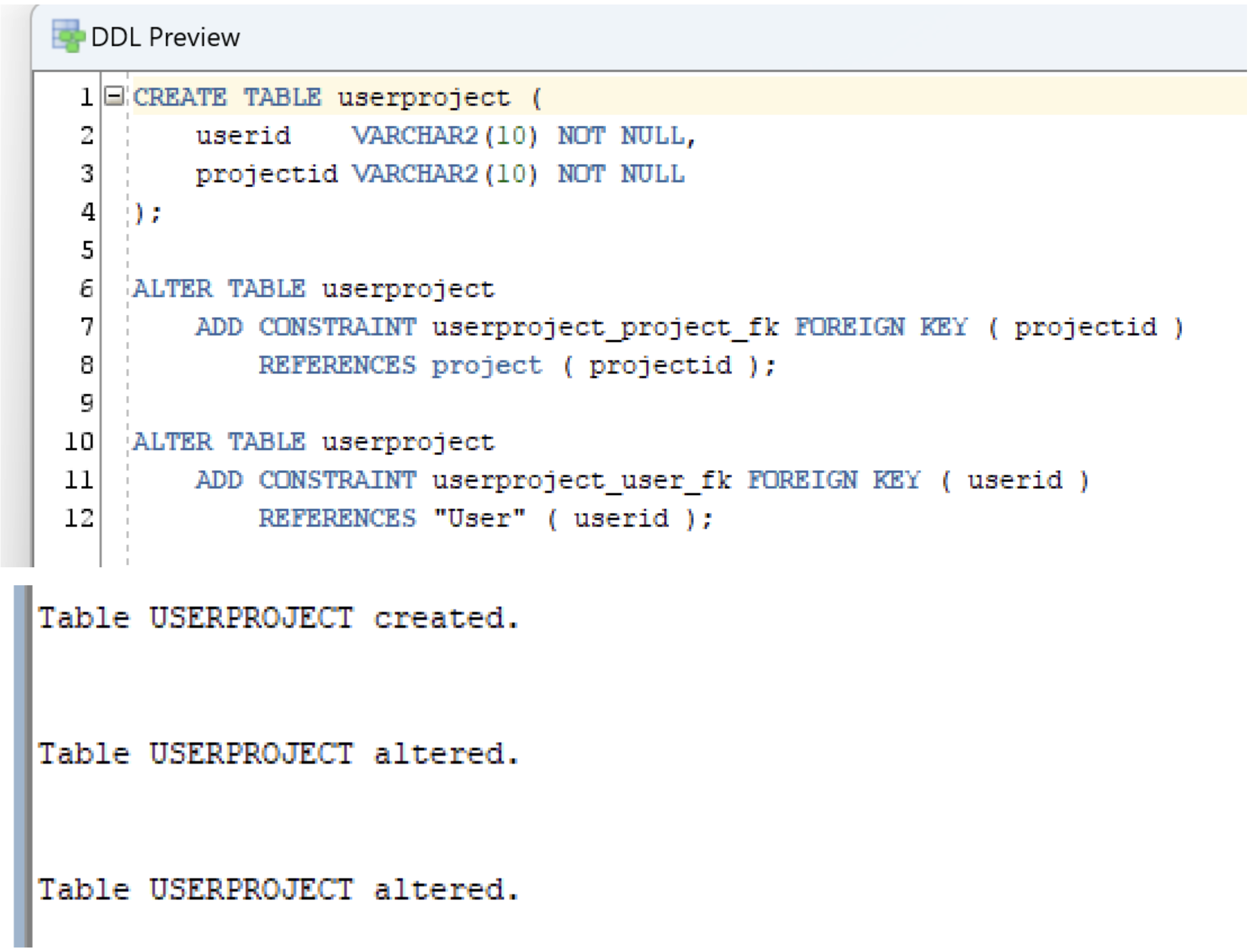


```
CREATE TABLE usertask (
    userid VARCHAR2(10) NOT NULL,
    taskid VARCHAR2(10) NOT NULL
);

ALTER TABLE usertask
    ADD CONSTRAINT usertask_task_fk FOREIGN KEY ( taskid )
        REFERENCES task ( taskid );

ALTER TABLE usertask
    ADD CONSTRAINT usertask_user_fk FOREIGN KEY ( userid )
        REFERENCES "User" ( userid );
```

```
Table USERTASK created.


Table USERTASK altered.


Table USERTASK altered.
```

Figure 9: UserTask Table

2.9 UserProject Table



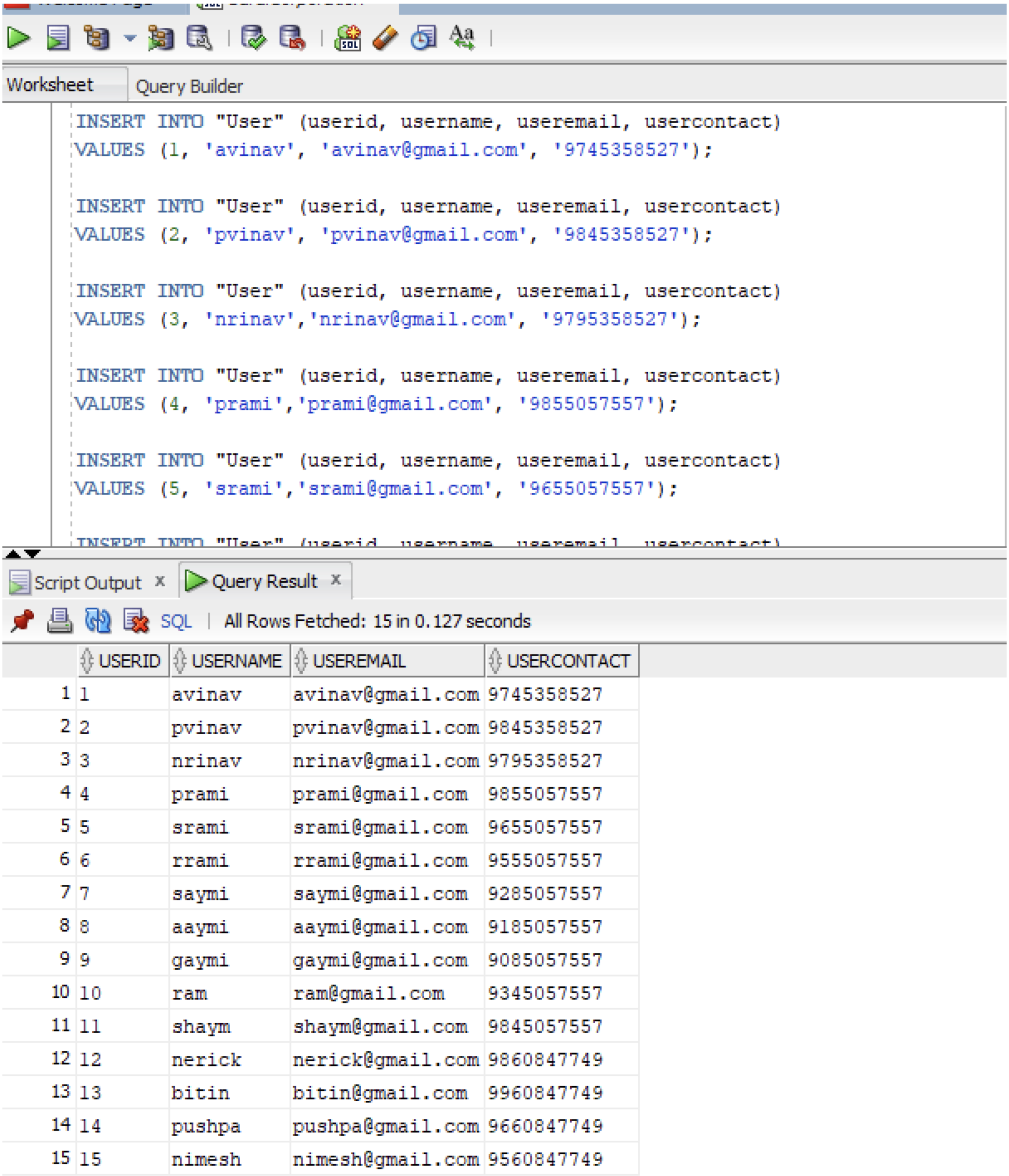
```
CREATE TABLE userproject (
    userid    VARCHAR2(10) NOT NULL,
    projectid VARCHAR2(10) NOT NULL
);

ALTER TABLE userproject
    ADD CONSTRAINT userproject_project_fk FOREIGN KEY ( projectid )
        REFERENCES project ( projectid );

ALTER TABLE userproject
    ADD CONSTRAINT userproject_user_fk FOREIGN KEY ( userid )
        REFERENCES "User" ( userid );
```

```
Table USERPROJECT created.


Table USERPROJECT altered.


Table USERPROJECT altered.
```

Figure 10: UserProject Table

1. Table Insert

3.1 User Table



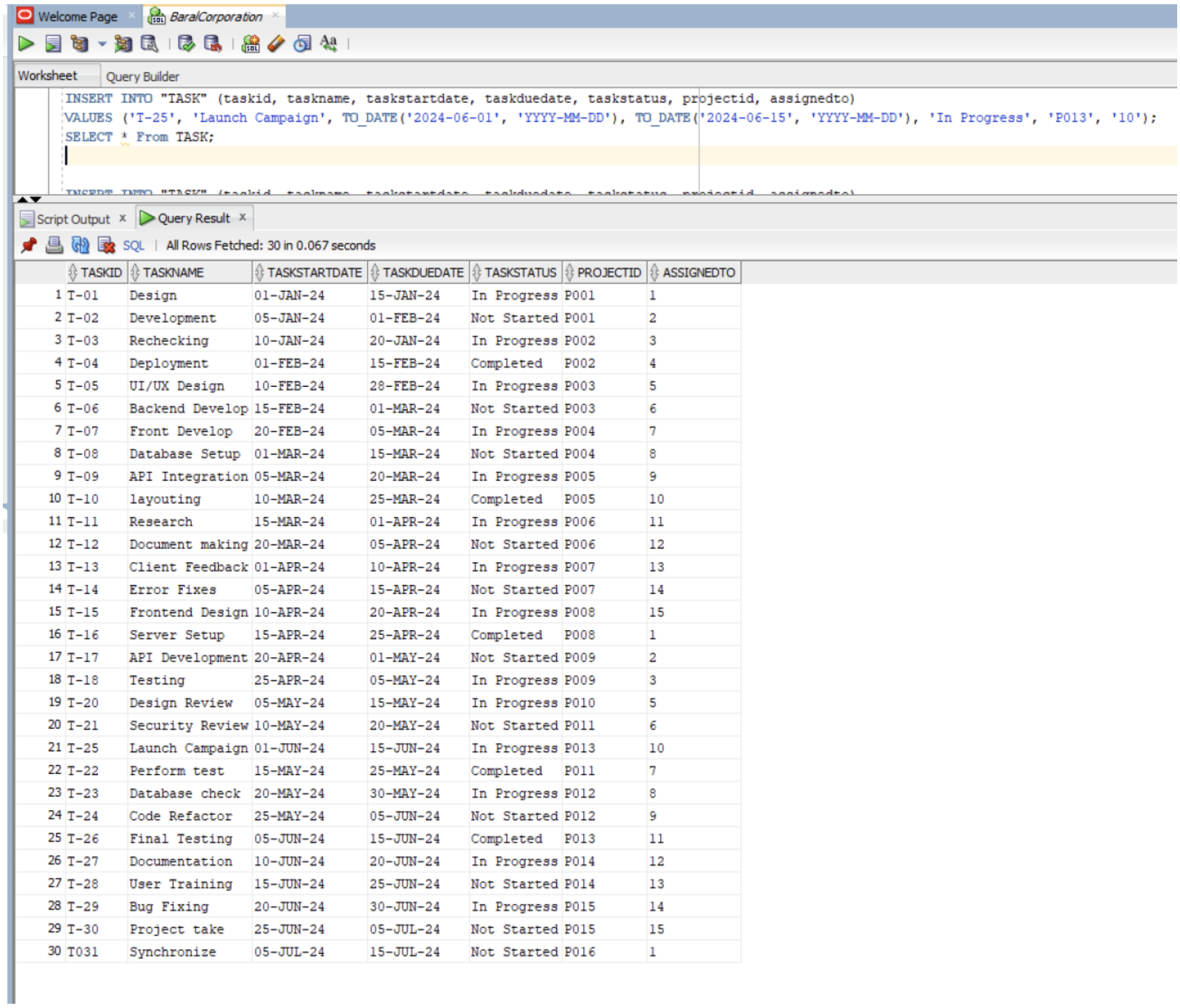
```
INSERT INTO "User" (userid, username, useremail, usercontact)
VALUES (1, 'avinav', 'avinav@gmail.com', '9745358527');

INSERT INTO "User" (userid, username, useremail, usercontact)
VALUES (2, 'pvinav', 'pvinav@gmail.com', '9845358527');

INSERT INTO "User" (userid, username, useremail, usercontact)
VALUES (3, 'nrinav','nrinav@gmail.com', '9795358527');

INSERT INTO "User" (userid, username, useremail, usercontact)
VALUES (4, 'prami','prami@gmail.com', '9855057557');

INSERT INTO "User" (userid, username, useremail, usercontact)
VALUES (5, 'srami','srami@gmail.com', '9655057557');

INSERT INTO "User" (userid, username, useremail, usercontact)
```

All Rows Fetched: 15 in 0.127 seconds

	USERID	USERNAME	USEREMAIL	USERCONTACT
1	1	avinav	avinav@gmail.com	9745358527
2	2	pvinav	pvinav@gmail.com	9845358527
3	3	nrinav	nrinav@gmail.com	9795358527
4	4	prami	prami@gmail.com	9855057557
5	5	srami	srami@gmail.com	9655057557
6	6	rrami	rrami@gmail.com	9555057557
7	7	saymi	saymi@gmail.com	9285057557
8	8	aaymi	aaymi@gmail.com	9185057557
9	9	gaymi	gaymi@gmail.com	9085057557
10	10	ram	ram@gmail.com	9345057557
11	11	shaym	shaym@gmail.com	9845057557
12	12	nerick	nerick@gmail.com	9860847749
13	13	bitin	bitin@gmail.com	9960847749
14	14	pushpa	pushpa@gmail.com	9660847749
15	15	nimesh	nimesh@gmail.com	9560847749

Figure 11: User Table value insert

3.2 Task Table

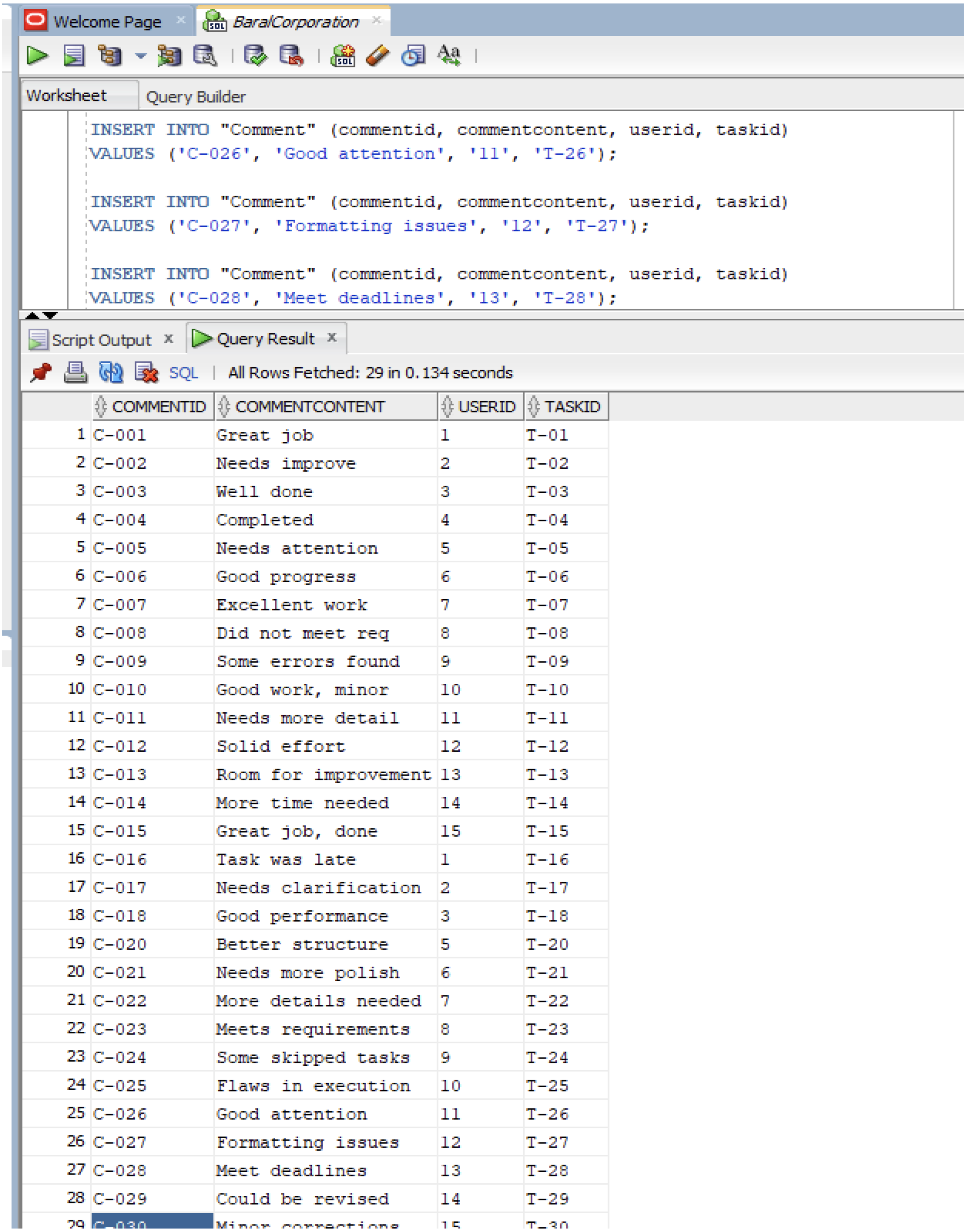


```
INSERT INTO "TASK" (taskid, taskname, taskstartdate, taskduedate, taskstatus, projectid, assignedto)
VALUES ('T-25', 'Launch Campaign', TO_DATE('2024-06-01', 'YYYY-MM-DD'), TO_DATE('2024-06-15', 'YYYY-MM-DD'), 'In Progress', 'P013', '10');
SELECT * From TASK;
```

	TASKID	TASKNAME	TASKSTARTDATE	TASKDUEDATE	TASKSTATUS	PROJECTID	ASSIGNEDTO
1	T-01	Design	01-JAN-24	15-JAN-24	In Progress	P001	1
2	T-02	Development	05-JAN-24	01-FEB-24	Not Started	P001	2
3	T-03	Rechecking	10-JAN-24	20-JAN-24	In Progress	P002	3
4	T-04	Deployment	01-FEB-24	15-FEB-24	Completed	P002	4
5	T-05	UI/UX Design	10-FEB-24	28-FEB-24	In Progress	P003	5
6	T-06	Backend Develop	15-FEB-24	01-MAR-24	Not Started	P003	6
7	T-07	Front Develop	20-FEB-24	05-MAR-24	In Progress	P004	7
8	T-08	Database Setup	01-MAR-24	15-MAR-24	Not Started	P004	8
9	T-09	API Integration	05-MAR-24	20-MAR-24	In Progress	P005	9
10	T-10	layouting	10-MAR-24	25-MAR-24	Completed	P005	10
11	T-11	Research	15-MAR-24	01-APR-24	In Progress	P006	11
12	T-12	Document making	20-MAR-24	05-APR-24	Not Started	P006	12
13	T-13	Client Feedback	01-APR-24	10-APR-24	In Progress	P007	13
14	T-14	Error Fixes	05-APR-24	15-APR-24	Not Started	P007	14
15	T-15	Frontend Design	10-APR-24	20-APR-24	In Progress	P008	15
16	T-16	Server Setup	15-APR-24	25-APR-24	Completed	P008	1
17	T-17	API Development	20-APR-24	01-MAY-24	Not Started	P009	2
18	T-18	Testing	25-APR-24	05-MAY-24	In Progress	P009	3
19	T-20	Design Review	05-MAY-24	15-MAY-24	In Progress	P010	5
20	T-21	Security Review	10-MAY-24	20-MAY-24	Not Started	P011	6
21	T-25	Launch Campaign	01-JUN-24	15-JUN-24	In Progress	P013	10
22	T-22	Perform test	15-MAY-24	25-MAY-24	Completed	P011	7
23	T-23	Database check	20-MAY-24	30-MAY-24	In Progress	P012	8
24	T-24	Code Refactor	25-MAY-24	05-JUN-24	Not Started	P012	9
25	T-26	Final Testing	05-JUN-24	15-JUN-24	Completed	P013	11
26	T-27	Documentation	10-JUN-24	20-JUN-24	In Progress	P014	12
27	T-28	User Training	15-JUN-24	25-JUN-24	Not Started	P014	13
28	T-29	Bug Fixing	20-JUN-24	30-JUN-24	In Progress	P015	14
29	T-30	Project take	25-JUN-24	05-JUL-24	Not Started	P015	15
30	T031	Synchronize	05-JUL-24	15-JUL-24	Not Started	P016	1

Figure 12: Task Table value insert

3.3 Comment Table



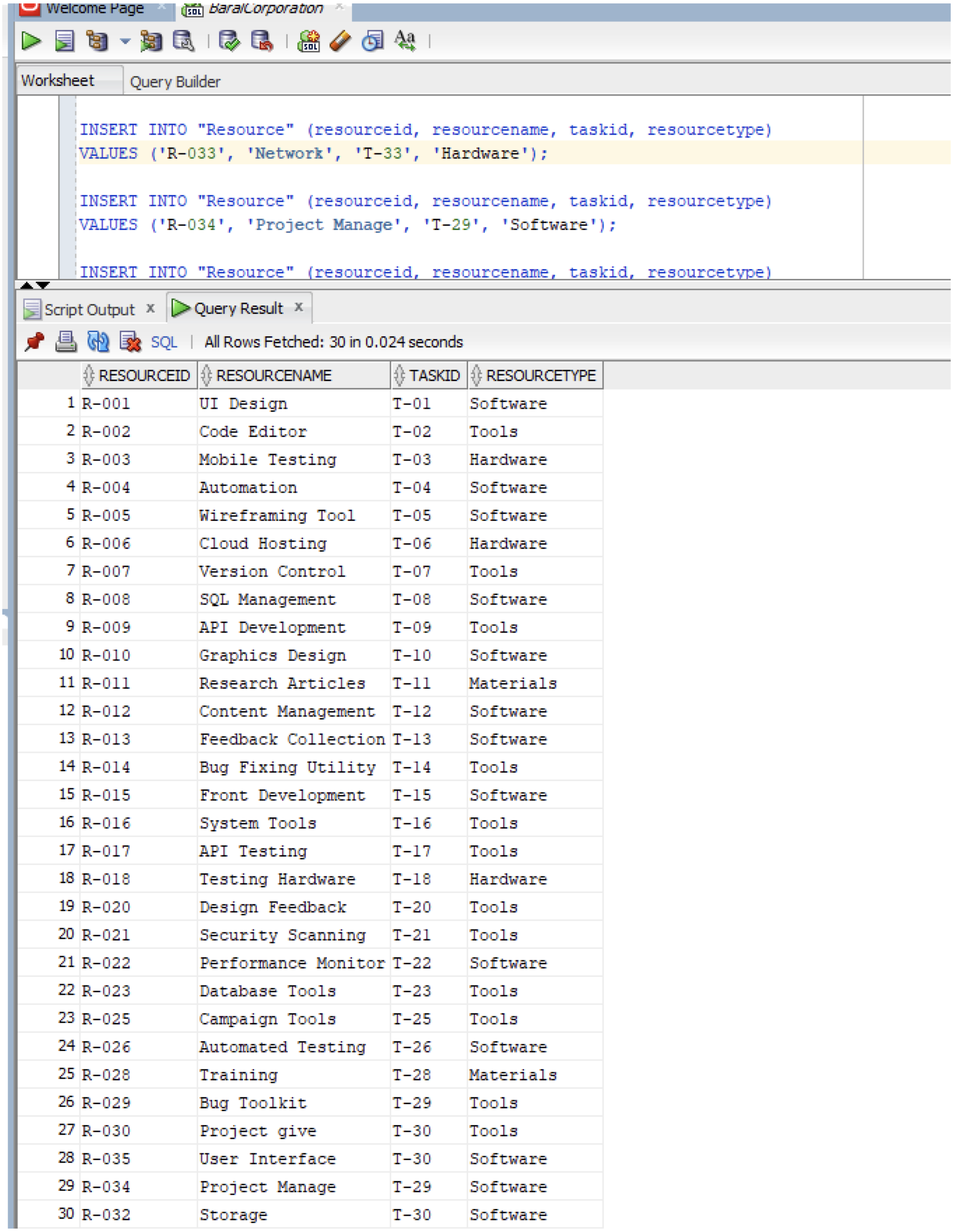
```
INSERT INTO "Comment" (commentid, commentcontent, userid, taskid)
VALUES ('C-026', 'Good attention', '11', 'T-26');

INSERT INTO "Comment" (commentid, commentcontent, userid, taskid)
VALUES ('C-027', 'Formatting issues', '12', 'T-27');

INSERT INTO "Comment" (commentid, commentcontent, userid, taskid)
VALUES ('C-028', 'Meet deadlines', '13', 'T-28');
```

All Rows Fetched: 29 in 0.134 seconds

	COMMENTID	COMMENTCONTENT	USERID	TASKID
1	C-001	Great job	1	T-01
2	C-002	Needs improve	2	T-02
3	C-003	Well done	3	T-03
4	C-004	Completed	4	T-04
5	C-005	Needs attention	5	T-05
6	C-006	Good progress	6	T-06
7	C-007	Excellent work	7	T-07
8	C-008	Did not meet req	8	T-08
9	C-009	Some errors found	9	T-09
10	C-010	Good work, minor	10	T-10
11	C-011	Needs more detail	11	T-11
12	C-012	Solid effort	12	T-12
13	C-013	Room for improvement	13	T-13
14	C-014	More time needed	14	T-14
15	C-015	Great job, done	15	T-15
16	C-016	Task was late	1	T-16
17	C-017	Needs clarification	2	T-17
18	C-018	Good performance	3	T-18
19	C-020	Better structure	5	T-20
20	C-021	Needs more polish	6	T-21
21	C-022	More details needed	7	T-22
22	C-023	Meets requirements	8	T-23
23	C-024	Some skipped tasks	9	T-24
24	C-025	Flaws in execution	10	T-25
25	C-026	Good attention	11	T-26
26	C-027	Formatting issues	12	T-27
27	C-028	Meet deadlines	13	T-28
28	C-029	Could be revised	14	T-29
29	C-030	Minor corrections	15	T-30

Figure 13: Comment Table value insert

3.4 Resource Table



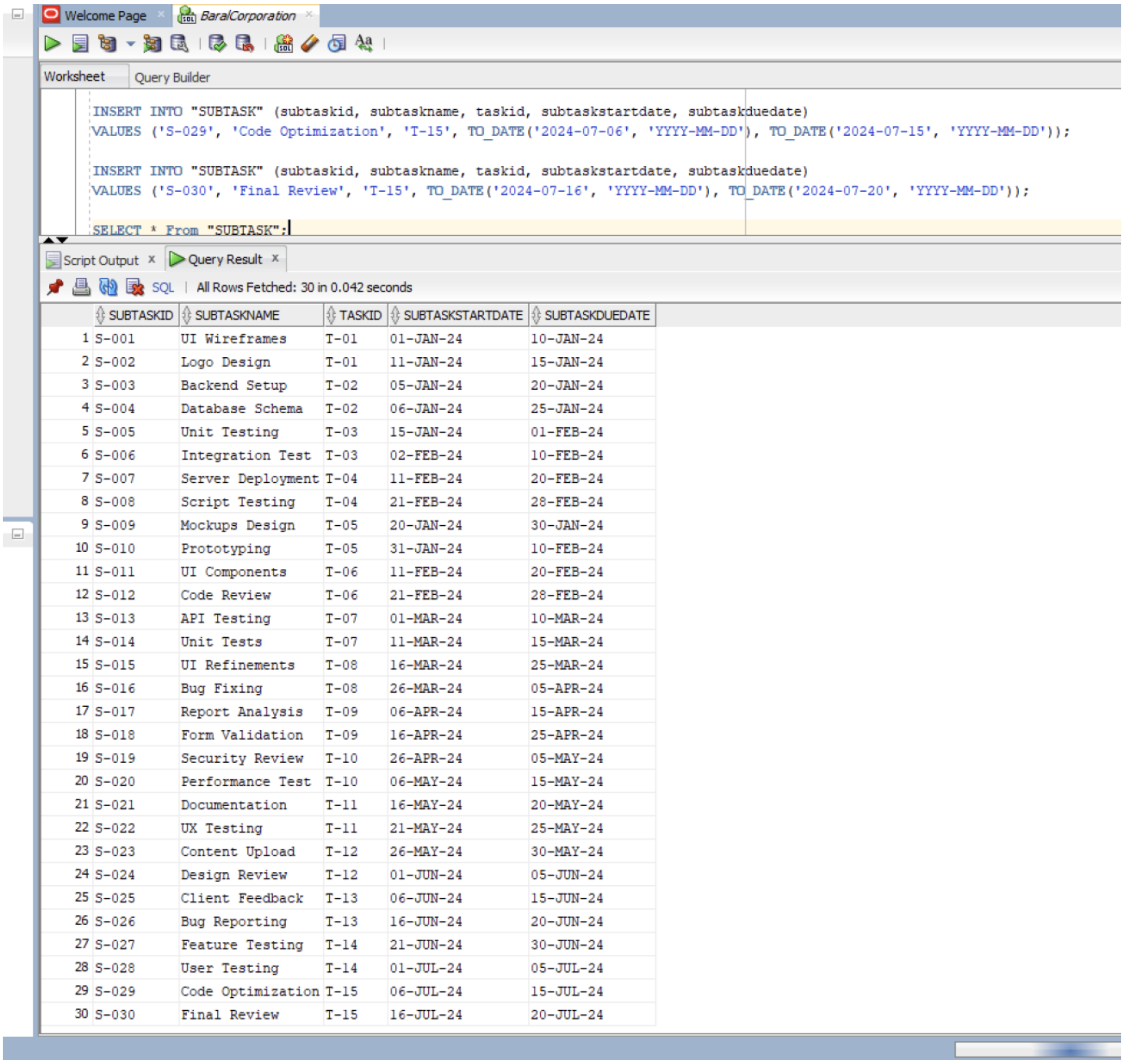
```
INSERT INTO "Resource" (resourceid, resourcename, taskid, resourcetype)
VALUES ('R-033', 'Network', 'T-33', 'Hardware');

INSERT INTO "Resource" (resourceid, resourcename, taskid, resourcetype)
VALUES ('R-034', 'Project Manage', 'T-29', 'Software');

INSERT INTO "Resource" (resourceid, resourcename, taskid, resourcetype)
```

All Rows Fetched: 30 in 0.024 seconds

	RESOURCEID	RESOURCENAME	TASKID	RESOURCETYPE
1	R-001	UI Design	T-01	Software
2	R-002	Code Editor	T-02	Tools
3	R-003	Mobile Testing	T-03	Hardware
4	R-004	Automation	T-04	Software
5	R-005	Wireframing Tool	T-05	Software
6	R-006	Cloud Hosting	T-06	Hardware
7	R-007	Version Control	T-07	Tools
8	R-008	SQL Management	T-08	Software
9	R-009	API Development	T-09	Tools
10	R-010	Graphics Design	T-10	Software
11	R-011	Research Articles	T-11	Materials
12	R-012	Content Management	T-12	Software
13	R-013	Feedback Collection	T-13	Software
14	R-014	Bug Fixing Utility	T-14	Tools
15	R-015	Front Development	T-15	Software
16	R-016	System Tools	T-16	Tools
17	R-017	API Testing	T-17	Tools
18	R-018	Testing Hardware	T-18	Hardware
19	R-020	Design Feedback	T-20	Tools
20	R-021	Security Scanning	T-21	Tools
21	R-022	Performance Monitor	T-22	Software
22	R-023	Database Tools	T-23	Tools
23	R-025	Campaign Tools	T-25	Tools
24	R-026	Automated Testing	T-26	Software
25	R-028	Training	T-28	Materials
26	R-029	Bug Toolkit	T-29	Tools
27	R-030	Project give	T-30	Tools
28	R-035	User Interface	T-30	Software
29	R-034	Project Manage	T-29	Software
30	R-032	Storage	T-30	Software

Figure 14: Resource Table value insert

3.5 Subtask Table



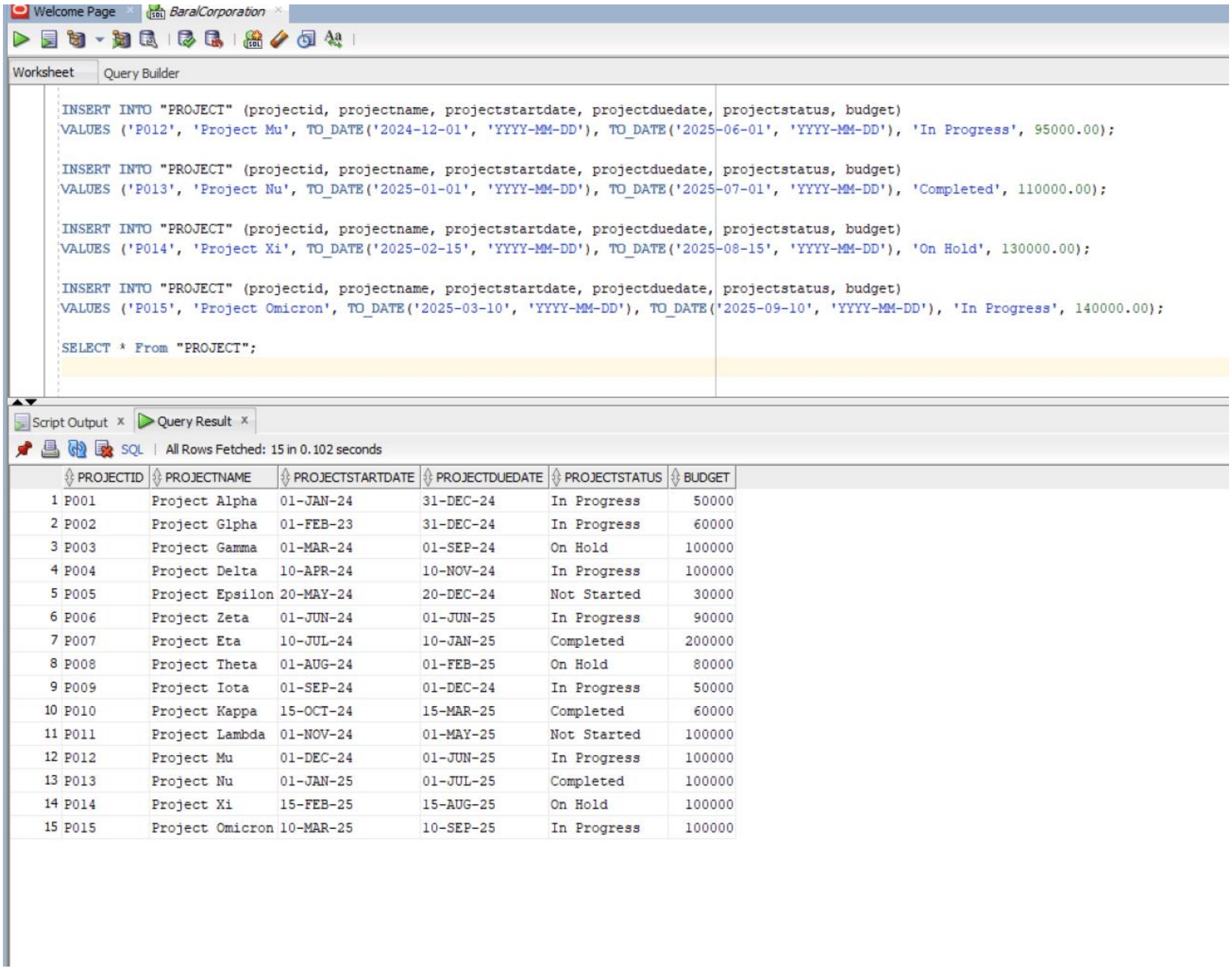
```
INSERT INTO "SUBTASK" (subtaskid, subtaskname, taskid, subtaskstartdate, subtaskduedate)
VALUES ('S-029', 'Code Optimization', 'T-15', TO_DATE('2024-07-06', 'YYYY-MM-DD'), TO_DATE('2024-07-15', 'YYYY-MM-DD'));

INSERT INTO "SUBTASK" (subtaskid, subtaskname, taskid, subtaskstartdate, subtaskduedate)
VALUES ('S-030', 'Final Review', 'T-15', TO_DATE('2024-07-16', 'YYYY-MM-DD'), TO_DATE('2024-07-20', 'YYYY-MM-DD'));

SELECT * From "SUBTASK";
```

All Rows Fetched: 30 in 0.042 seconds

	SUBTASKID	SUBTASKNAME	TASKID	SUBTASKSTARTDATE	SUBTASKDUEDATE
1	S-001	UI Wireframes	T-01	01-JAN-24	10-JAN-24
2	S-002	Logo Design	T-01	11-JAN-24	15-JAN-24
3	S-003	Backend Setup	T-02	05-JAN-24	20-JAN-24
4	S-004	Database Schema	T-02	06-JAN-24	25-JAN-24
5	S-005	Unit Testing	T-03	15-JAN-24	01-FEB-24
6	S-006	Integration Test	T-03	02-FEB-24	10-FEB-24
7	S-007	Server Deployment	T-04	11-FEB-24	20-FEB-24
8	S-008	Script Testing	T-04	21-FEB-24	28-FEB-24
9	S-009	Mockups Design	T-05	20-JAN-24	30-JAN-24
10	S-010	Prototyping	T-05	31-JAN-24	10-FEB-24
11	S-011	UI Components	T-06	11-FEB-24	20-FEB-24
12	S-012	Code Review	T-06	21-FEB-24	28-FEB-24
13	S-013	API Testing	T-07	01-MAR-24	10-MAR-24
14	S-014	Unit Tests	T-07	11-MAR-24	15-MAR-24
15	S-015	UI Refinements	T-08	16-MAR-24	25-MAR-24
16	S-016	Bug Fixing	T-08	26-MAR-24	05-APR-24
17	S-017	Report Analysis	T-09	06-APR-24	15-APR-24
18	S-018	Form Validation	T-09	16-APR-24	25-APR-24
19	S-019	Security Review	T-10	26-APR-24	05-MAY-24
20	S-020	Performance Test	T-10	06-MAY-24	15-MAY-24
21	S-021	Documentation	T-11	16-MAY-24	20-MAY-24
22	S-022	UX Testing	T-11	21-MAY-24	25-MAY-24
23	S-023	Content Upload	T-12	26-MAY-24	30-MAY-24
24	S-024	Design Review	T-12	01-JUN-24	05-JUN-24
25	S-025	Client Feedback	T-13	06-JUN-24	15-JUN-24
26	S-026	Bug Reporting	T-13	16-JUN-24	20-JUN-24
27	S-027	Feature Testing	T-14	21-JUN-24	30-JUN-24
28	S-028	User Testing	T-14	01-JUL-24	05-JUL-24
29	S-029	Code Optimization	T-15	06-JUL-24	15-JUL-24
30	S-030	Final Review	T-15	16-JUL-24	20-JUL-24

Figure 15: Subtask table value insert

3.6 Project Table



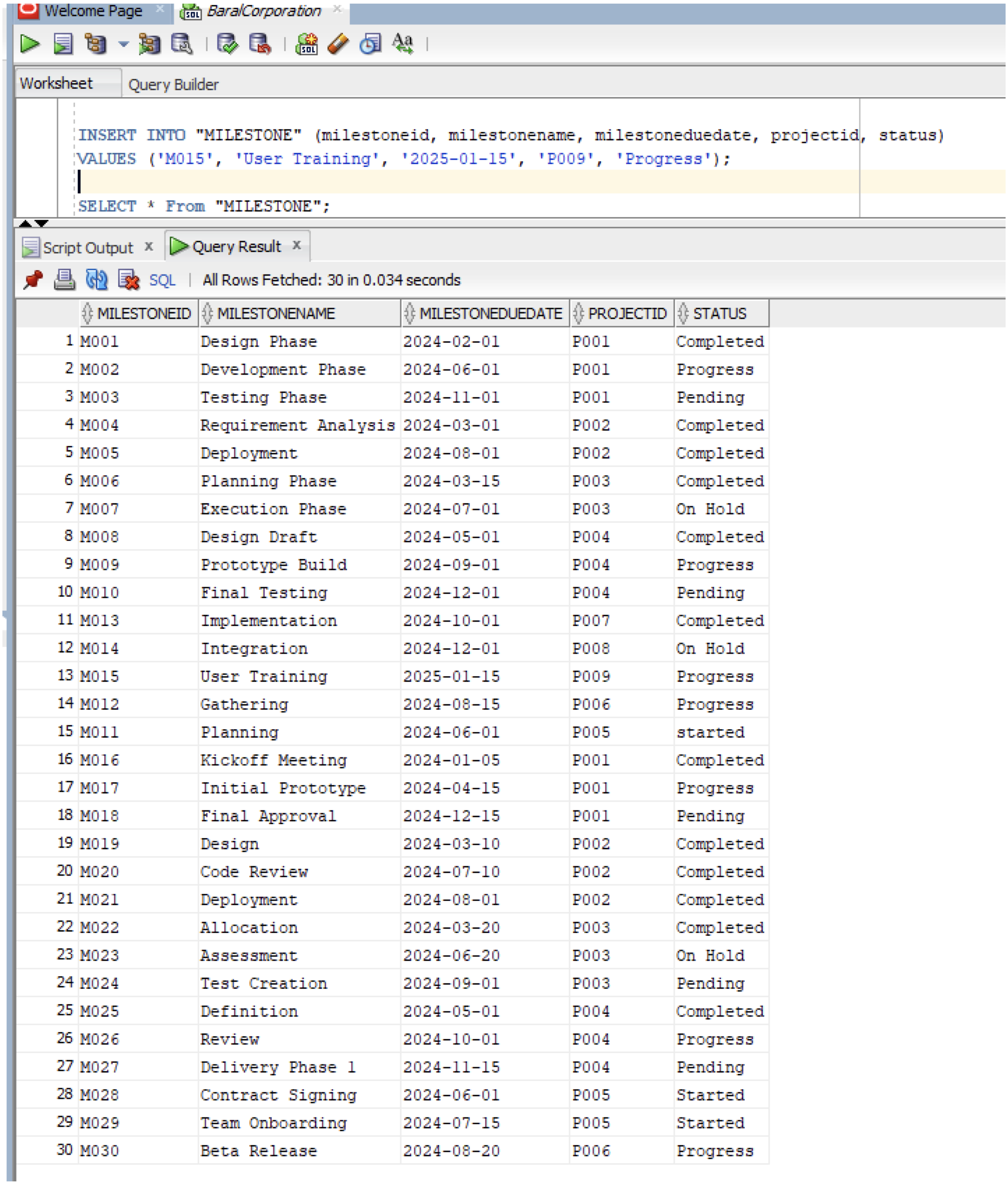
```
INSERT INTO "PROJECT" (projectid, projectname, projectstartdate, projectduedate, projectstatus, budget)
VALUES ('P012', 'Project Mu', TO_DATE('2024-12-01', 'YYYY-MM-DD'), TO_DATE('2025-06-01', 'YYYY-MM-DD'), 'In Progress', 95000.00);

INSERT INTO "PROJECT" (projectid, projectname, projectstartdate, projectduedate, projectstatus, budget)
VALUES ('P013', 'Project Nu', TO_DATE('2025-01-01', 'YYYY-MM-DD'), TO_DATE('2025-07-01', 'YYYY-MM-DD'), 'Completed', 110000.00);

INSERT INTO "PROJECT" (projectid, projectname, projectstartdate, projectduedate, projectstatus, budget)
VALUES ('P014', 'Project Xi', TO_DATE('2025-02-15', 'YYYY-MM-DD'), TO_DATE('2025-08-15', 'YYYY-MM-DD'), 'On Hold', 130000.00);

INSERT INTO "PROJECT" (projectid, projectname, projectstartdate, projectduedate, projectstatus, budget)
VALUES ('P015', 'Project Omicron', TO_DATE('2025-03-10', 'YYYY-MM-DD'), TO_DATE('2025-09-10', 'YYYY-MM-DD'), 'In Progress', 140000.00);

SELECT * From "PROJECT";
```

All Rows Fetched: 15 in 0.102 seconds

	PROJECTID	PROJECTNAME	PROJECTSTARTDATE	PROJECTDUEDATE	PROJECTSTATUS	BUDGET
1	P001	Project Alpha	01-JAN-24	31-DEC-24	In Progress	50000
2	P002	Project Glpha	01-FEB-23	31-DEC-24	In Progress	60000
3	P003	Project Gamma	01-MAR-24	01-SEP-24	On Hold	100000
4	P004	Project Delta	10-APR-24	10-NOV-24	In Progress	100000
5	P005	Project Epsilon	20-MAY-24	20-DEC-24	Not Started	30000
6	P006	Project Zeta	01-JUN-24	01-JUN-25	In Progress	90000
7	P007	Project Eta	10-JUL-24	10-JAN-25	Completed	200000
8	P008	Project Theta	01-AUG-24	01-FEB-25	On Hold	80000
9	P009	Project Iota	01-SEP-24	01-DEC-24	In Progress	50000
10	P010	Project Kappa	15-OCT-24	15-MAR-25	Completed	60000
11	P011	Project Lambda	01-NOV-24	01-MAY-25	Not Started	100000
12	P012	Project Mu	01-DEC-24	01-JUN-25	In Progress	100000
13	P013	Project Nu	01-JAN-25	01-JUL-25	Completed	100000
14	P014	Project Xi	15-FEB-25	15-AUG-25	On Hold	100000
15	P015	Project Omicron	10-MAR-25	10-SEP-25	In Progress	100000

Figure 16: Project Table value insert

3.7 Milestone Table



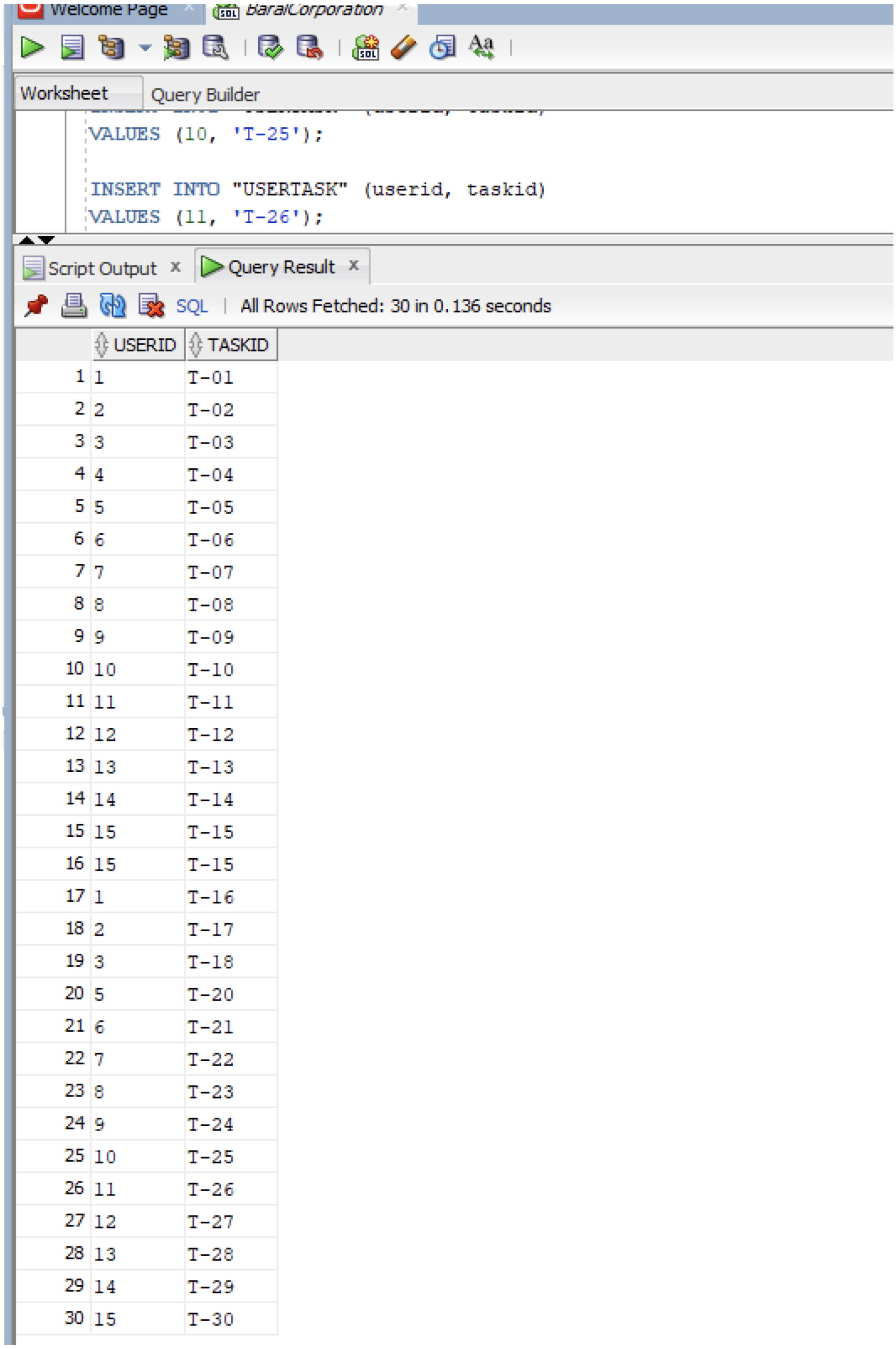
```
INSERT INTO "MILESTONE" (milestoneid, milestonename, milestoneduedate, projectid, status)
VALUES ('M015', 'User Training', '2025-01-15', 'P009', 'Progress');

SELECT * From "MILESTONE";
```

All Rows Fetched: 30 in 0.034 seconds

	MILESTONEID	MILESTONENAME	MILESTONEDUEDATE	PROJECTID	STATUS
1	M001	Design Phase	2024-02-01	P001	Completed
2	M002	Development Phase	2024-06-01	P001	Progress
3	M003	Testing Phase	2024-11-01	P001	Pending
4	M004	Requirement Analysis	2024-03-01	P002	Completed
5	M005	Deployment	2024-08-01	P002	Completed
6	M006	Planning Phase	2024-03-15	P003	Completed
7	M007	Execution Phase	2024-07-01	P003	On Hold
8	M008	Design Draft	2024-05-01	P004	Completed
9	M009	Prototype Build	2024-09-01	P004	Progress
10	M010	Final Testing	2024-12-01	P004	Pending
11	M013	Implementation	2024-10-01	P007	Completed
12	M014	Integration	2024-12-01	P008	On Hold
13	M015	User Training	2025-01-15	P009	Progress
14	M012	Gathering	2024-08-15	P006	Progress
15	M011	Planning	2024-06-01	P005	started
16	M016	Kickoff Meeting	2024-01-05	P001	Completed
17	M017	Initial Prototype	2024-04-15	P001	Progress
18	M018	Final Approval	2024-12-15	P001	Pending
19	M019	Design	2024-03-10	P002	Completed
20	M020	Code Review	2024-07-10	P002	Completed
21	M021	Deployment	2024-08-01	P002	Completed
22	M022	Allocation	2024-03-20	P003	Completed
23	M023	Assessment	2024-06-20	P003	On Hold
24	M024	Test Creation	2024-09-01	P003	Pending
25	M025	Definition	2024-05-01	P004	Completed
26	M026	Review	2024-10-01	P004	Progress
27	M027	Delivery Phase 1	2024-11-15	P004	Pending
28	M028	Contract Signing	2024-06-01	P005	Started
29	M029	Team Onboarding	2024-07-15	P005	Started
30	M030	Beta Release	2024-08-20	P006	Progress

Figure 17: Milestone Table value insert

3.8 UserTask Table



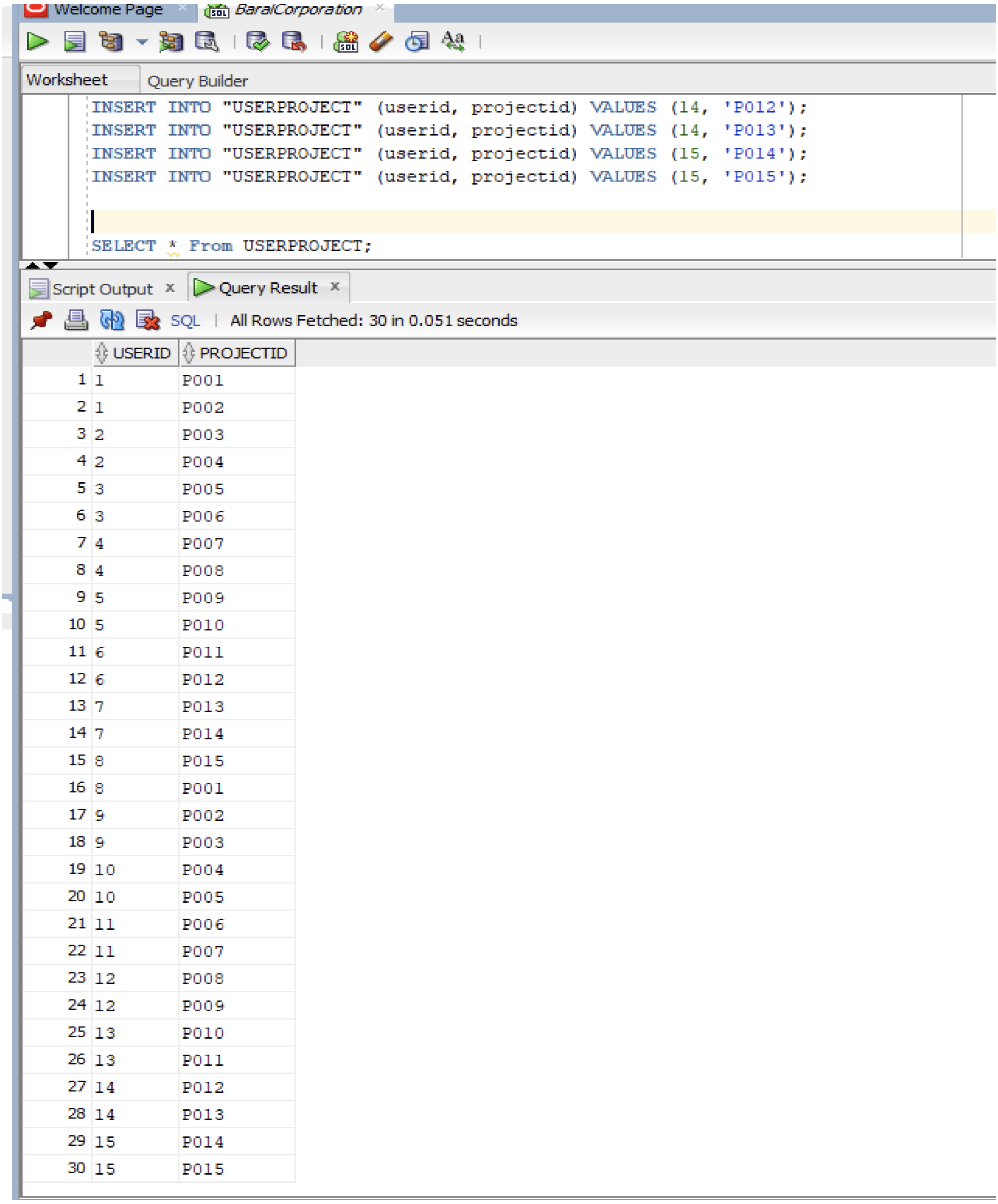
```
VALUES (10, 'T-25');

INSERT INTO "USERTASK" (userid, taskid)
VALUES (11, 'T-26');
```

All Rows Fetched: 30 in 0.136 seconds

	USERID	TASKID
1	1	T-01
2	2	T-02
3	3	T-03
4	4	T-04
5	5	T-05
6	6	T-06
7	7	T-07
8	8	T-08
9	9	T-09
10	10	T-10
11	11	T-11
12	12	T-12
13	13	T-13
14	14	T-14
15	15	T-15
16	15	T-15
17	1	T-16
18	2	T-17
19	3	T-18
20	5	T-20
21	6	T-21
22	7	T-22
23	8	T-23
24	9	T-24
25	10	T-25
26	11	T-26
27	12	T-27
28	13	T-28
29	14	T-29
30	15	T-30

Figure 18: UserTask Table value insert

3.9 UserProject Table



```
INSERT INTO "USERPROJECT" (userid, projectid) VALUES (14, 'P012');
INSERT INTO "USERPROJECT" (userid, projectid) VALUES (14, 'P013');
INSERT INTO "USERPROJECT" (userid, projectid) VALUES (15, 'P014');
INSERT INTO "USERPROJECT" (userid, projectid) VALUES (15, 'P015');

SELECT * From USERPROJECT;
```

All Rows Fetched: 30 in 0.051 seconds

	USERID	PROJECTID
1	1	P001
2	1	P002
3	2	P003
4	2	P004
5	3	P005
6	3	P006
7	4	P007
8	4	P008
9	5	P009
10	5	P010
11	6	P011
12	6	P012
13	7	P013
14	7	P014
15	8	P015
16	8	P001
17	9	P002
18	9	P003
19	10	P004
20	10	P005
21	11	P006
22	11	P007
23	12	P008
24	12	P009
25	13	P010
26	13	P011
27	14	P012
28	14	P013
29	15	P014
30	15	P015

Figure 19:UserProject Table