Schedule Management System

Pragati Brijesh Shukla
Rashmi Chaudhari
Siddharth Rawat
Soumya Roy
Vivekkumar Singh

Table of Contents

ABSTRACT **3**

INTRODUCTION **3**

THE IMPORTANCE OF COURSE SCHEDULING **3**

WEIGHT DISTRIBUTION 4

ASSUMPTIONS **4**

ENTITY RELATIONSHIP DIAGRAM **5**

DATABASE DESIGN **6**

ENTITIES AND ATTRIBUTES: 6

DATA GENERATION AND LOADING **7**

COURSES: 7

DEPARTMENT: 8

SECTION: 8

LOCATION: 9

INSTRUCTOR: 9

TIMETABLE: 10

TABLE STRUCTURES **11**

COURSES: 11

DEPARTMENT: 11

SECTION: 11

LOCATION: 11

INSTRUCTOR: 12

TIMETABLE: 12

DATA INSERTION **13**

USING QUERIES (ONLY A PART IS INCLUDED IN THE SCREENSHOT) 13

IMPORTING DATA USING CSV/EXCEL FILES 14

DATA GENERATION **15**

SCREENSHOTS OF ACTUAL DATA IN TABLES: 16

PERFORMANCE TUNING **22**

INDEXING 22

SAMPLE QUERIES 24

DATA VISUALIZATION **26**

FUTURE SCOPE **27**

ABSTRACT

Universities and colleges face issues while scheduling since on every day of the week the classes offered are different, belong to different departments and may/may not be offered in a semester. Thus, Schedule management system is a must to solve the challenge of scheduling. This scheduler will help manage the class timetables offered by instructors. One instructor will not have conflicting classes to offer, students will be able to decide and plan their schedule by deciding on the instructor for their desired course.

INTRODUCTION

Scheduler system helps to manage the class schedules of the university depending on the department and courses. It keeps the record of instructors who teach courses and their timetables. Any student, professor or advisor across all departments of the University can check the schedule of any course and other details like the location of class, time, date and so on.

Every instructor has a unique instructor ID and he/she belongs to a department having unique department ID. Instructors from each department will have courses and class timings. This will help in organizing classes and letting the students select courses, which don't have conflicting timings. It will serve in utilizing the time of each semester efficiently.

THE IMPORTANCE OF COURSE SCHEDULING

Course schedules can impact a wide range of institutional outcomes, including student satisfaction, graduation rates and time to graduation, and university expenditures. For example, in a 2012 national student satisfaction survey, respondents indicated that course availability was a top concern, especially for nontraditional students (e.g., over 25 years of age and those who are employed full-time).

Similarly, university administrators regularly note that "managing the schedule with students' needs in mind is critical for retention. Given the importance that scheduling can have on how students, faculty, and administrators make decisions, it becomes essential that course catalogs reflect student and institutional needs. For instance, more than one-third of entry-level courses at four-year, public universities are over-enrolled, suggesting that the availability of these courses needs to be expanded. However, institutions of higher education seemingly do not regularly re-examine course schedules to reflect changes in student demand or institutional capacities.

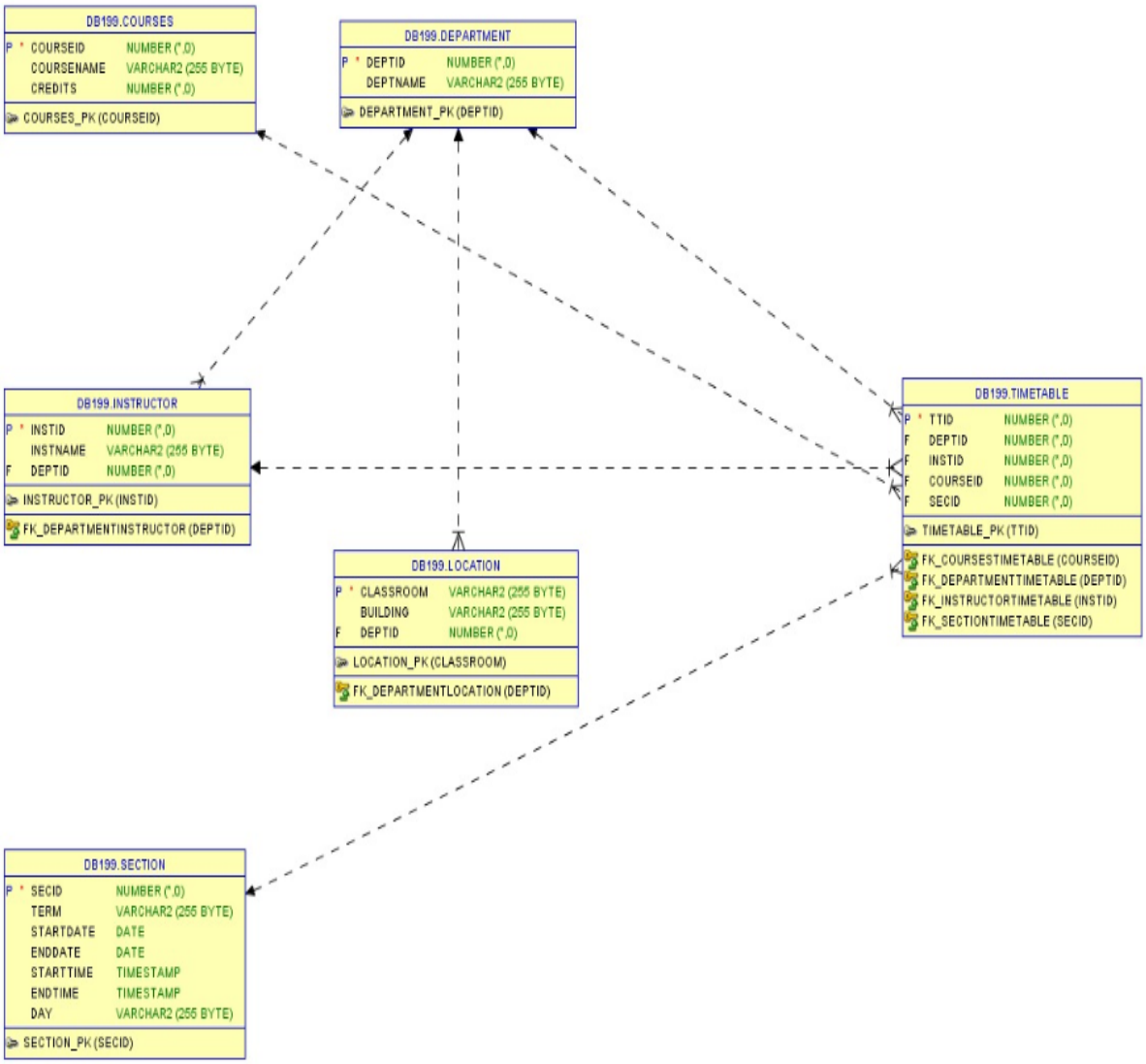
Weight Distribution

Topic Area	Points
Database Design	30
Query Writing	30
Performance Tuning	20
Others	20

Assumptions

1. All Schedules are readily available in advance with the administrator
2. There are no mismatching of course and course ID
3. Each course has a designated professor
4. A student can be assigned only a limited number courses as per the college rules in a single semester
5. A scheduler can create a course schedule only for the oncoming semester

Entity Relationship Diagram



Database Design

Entities and Attributes:

Table Name	Attributes
Courses	Courseid, Coursename, Instid, Credits
Department	Deptid, Deptname
Section	Secid, Term, Startdate, Enddate, Day
Location	Classroom, Building, Deptid
Instructor	Instid, Instname, Deptid
Timetable	TTid, Deptid, Instid, Courseid, Secid

Data Generation and Loading

Creating the Tables Using Query:

Courses:

```
CREATE TABLE "DB199"."COURSES1"
 (    "COURSEID" NUMBER(*,0) NOT NULL ENABLE,
  "COURSENAME" VARCHAR2(255 BYTE),
  "INSTID" NUMBER(*,0) NOT NULL ENABLE,
  "CREDITS" NUMBER(*,0),
   PRIMARY KEY ("COURSEID")
USING INDEX PCTFREE 10 INITRANS 2 MAXTRANS 255 COMPUTE STATISTICS
STORAGE(INITIAL 65536 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645
PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1
BUFFER_POOL DEFAULT FLASH_CACHE DEFAULT CELL_FLASH_CACHE DEFAULT)
TABLESPACE "STUDENTS"  ENABLE
 ) SEGMENT CREATION IMMEDIATE
PCTFREE 10 PCTUSED 40 INITRANS 1 MAXTRANS 255
NOCOMPRESS LOGGING
STORAGE(INITIAL 65536 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645
PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1
BUFFER_POOL DEFAULT FLASH_CACHE DEFAULT CELL_FLASH_CACHE DEFAULT)
TABLESPACE "STUDENTS" ;
```

Department:

```
 CREATE TABLE "DB199"."DEPARTMENT1"
  (    "DEPTID" NUMBER(*,0) NOT NULL ENABLE,
   "DEPTNAME" VARCHAR2(255 BYTE),
    PRIMARY KEY ("DEPTID")
 USING INDEX PCTFREE 10 INITRANS 2 MAXTRANS 255 COMPUTE STATISTICS
 STORAGE(INITIAL 65536 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645
 PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1
 BUFFER_POOL DEFAULT FLASH_CACHE DEFAULT CELL_FLASH_CACHE DEFAULT)
 TABLESPACE "STUDENTS"  ENABLE
  ) SEGMENT CREATION IMMEDIATE
 PCTFREE 10 PCTUSED 40 INITRANS 1 MAXTRANS 255
NOCOMPRESS LOGGING
 STORAGE(INITIAL 65536 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645
 PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1
 BUFFER_POOL DEFAULT FLASH_CACHE DEFAULT CELL_FLASH_CACHE DEFAULT)
 TABLESPACE "STUDENTS" ;
```

Section:

```
 CREATE TABLE "DB199"."SECTION1"
  (    "SECID" NUMBER(*,0) NOT NULL ENABLE,
   "TERM" NUMBER(*,0),
   "STARTDATE" VARCHAR2(255 BYTE),
   "ENDDATE" VARCHAR2(255 BYTE),
   "DAY" NUMBER(*,0),
    PRIMARY KEY ("SECID")
 USING INDEX PCTFREE 10 INITRANS 2 MAXTRANS 255 COMPUTE STATISTICS
 STORAGE(INITIAL 65536 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645
 PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1
 BUFFER_POOL DEFAULT FLASH_CACHE DEFAULT CELL_FLASH_CACHE DEFAULT)
 TABLESPACE "STUDENTS"  ENABLE
  ) SEGMENT CREATION IMMEDIATE
 PCTFREE 10 PCTUSED 40 INITRANS 1 MAXTRANS 255
NOCOMPRESS LOGGING
 STORAGE(INITIAL 65536 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645
 PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1
 BUFFER_POOL DEFAULT FLASH_CACHE DEFAULT CELL_FLASH_CACHE DEFAULT)
 TABLESPACE "STUDENTS" ;
```

Location:

```
CREATE TABLE "DB199"."LOCATION1"
 (    "CLASSROOM" VARCHAR2(255 BYTE) NOT NULL ENABLE,
  "BUILDING" VARCHAR2(255 BYTE),
  "DEPTID" NUMBER(*,0),
   PRIMARY KEY ("CLASSROOM")
USING INDEX PCTFREE 10 INITRANS 2 MAXTRANS 255 COMPUTE STATISTICS
STORAGE(INITIAL 65536 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645
PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1
BUFFER_POOL DEFAULT FLASH_CACHE DEFAULT CELL_FLASH_CACHE DEFAULT)
TABLESPACE "STUDENTS"  ENABLE,
   CONSTRAINT "FK_DEPARTMENTLOCATION1" FOREIGN KEY ("DEPTID")
    REFERENCES "DB199"."DEPARTMENT" ("DEPTID") ENABLE
 ) SEGMENT CREATION IMMEDIATE
PCTFREE 10 PCTUSED 40 INITRANS 1 MAXTRANS 255
NOCOMPRESS LOGGING
STORAGE(INITIAL 65536 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645
PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1
BUFFER_POOL DEFAULT FLASH_CACHE DEFAULT CELL_FLASH_CACHE DEFAULT)
TABLESPACE "STUDENTS" ;
```

Instructor:

```
CREATE TABLE "DB199"."INSTRUCTOR1"
 (    "INSTID" VARCHAR2(50 BYTE),
  "INSTNAME" VARCHAR2(255 BYTE) DEFAULT NULL,
  "DEPTID" VARCHAR2(50 BYTE) DEFAULT NULL,
   PRIMARY KEY ("INSTID")
USING INDEX PCTFREE 10 INITRANS 2 MAXTRANS 255 COMPUTE STATISTICS
STORAGE(INITIAL 65536 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645
PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1
BUFFER_POOL DEFAULT FLASH_CACHE DEFAULT CELL_FLASH_CACHE DEFAULT)
TABLESPACE "STUDENTS"  ENABLE
 ) SEGMENT CREATION IMMEDIATE
PCTFREE 10 PCTUSED 40 INITRANS 1 MAXTRANS 255
NOCOMPRESS LOGGING
STORAGE(INITIAL 65536 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645
PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1
BUFFER_POOL DEFAULT FLASH_CACHE DEFAULT CELL_FLASH_CACHE DEFAULT)
TABLESPACE "STUDENTS" ;
```

Timetable:

```
 CREATE TABLE "DB199"."TIMETABLE3"
  (    "TTID" NUMBER(*,0) NOT NULL ENABLE,
   "DEPTID" NUMBER(*,0),
   "INSTID" NUMBER(*,0),
   "COURSEID" NUMBER(*,0),
   "SECID" NUMBER(*,0),
    PRIMARY KEY ("TTID")
 USING INDEX PCTFREE 10 INITRANS 2 MAXTRANS 255 COMPUTE STATISTICS
 STORAGE(INITIAL 65536 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645
 PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1
 BUFFER_POOL DEFAULT FLASH_CACHE DEFAULT CELL_FLASH_CACHE DEFAULT)
 TABLESPACE "STUDENTS"  ENABLE
  ) SEGMENT CREATION IMMEDIATE
 PCTFREE 10 PCTUSED 40 INITRANS 1 MAXTRANS 255
NOCOMPRESS LOGGING
 STORAGE(INITIAL 65536 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645
 PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1
 BUFFER_POOL DEFAULT FLASH_CACHE DEFAULT CELL_FLASH_CACHE DEFAULT)
 TABLESPACE "STUDENTS" ;
```

TABLE STRUCTURES

Courses:

	COLUMN_NAME	DATA_TYPE	NULLABLE	DATA_DEFAULT	COLUMN_ID	COMMENTS
1	COURSEID	NUMBER(38,0)	No	(null)	1	(null)
2	COURSENAME	VARCHAR2(255 BYTE)	Yes	(null)	2	(null)
3	INSTID	NUMBER(38,0)	No	(null)	3	(null)
4	CREDITS	NUMBER(38,0)	Yes	(null)	4	(null)

Department:

	COLUMN_NAME	DATA_TYPE	NULLABLE	DATA_DEFAULT	COLUMN_ID	COMMENTS
1	DEPTID	NUMBER(38,0)	No	(null)	1	(null)
2	DEPTNAME	VARCHAR2(255 BYTE)	Yes	(null)	2	(null)

Section:

	COLUMN_NAME	DATA_TYPE	NULLABLE	DATA_DEFAULT	COLUMN_ID	COMMENTS
1	SECID	NUMBER(38,0)	No	(null)	1	(null)
2	TERM	NUMBER(38,0)	Yes	(null)	2	(null)
3	STARTDATE	VARCHAR2(255 BYTE)	Yes	(null)	3	(null)
4	ENDDATE	VARCHAR2(255 BYTE)	Yes	(null)	4	(null)
5	DAY	NUMBER(38,0)	Yes	(null)	5	(null)

Location:

	COLUMN_NAME	DATA_TYPE	NULLABLE	DATA_DEFAULT	COLUMN_ID	COMMENTS
1	CLASSROOM	VARCHAR2(255 BYTE)	No	(null)	1	(null)
2	BUILDING	VARCHAR2(255 BYTE)	Yes	(null)	2	(null)
3	DEPTID	NUMBER(38,0)	Yes	(null)	3	(null)

Instructor:

	COLUMN_NAME	DATA_TYPE	NULLABLE	DATA_DEFAULT	COLUMN_ID	COMMENTS
1	INSTID	VARCHAR2(50 BYTE)	No	(null)	1	(null)
2	INSTNAME	VARCHAR2(255 BYTE)	Yes	NULL	2	(null)
3	DEPTID	VARCHAR2(50 BYTE)	Yes	NULL	3	(null)

Timetable:

	COLUMN_NAME	DATA_TYPE	NULLABLE	DATA_DEFAULT	COLUMN_ID	COMMENTS
1	TTID	NUMBER(38,0)	No	(null)	1	(null)
2	DEPTID	NUMBER(38,0)	Yes	(null)	2	(null)
3	INSTID	NUMBER(38,0)	Yes	(null)	3	(null)
4	COURSEID	NUMBER(38,0)	Yes	(null)	4	(null)
5	SECID	NUMBER(38,0)	Yes	(null)	5	(null)

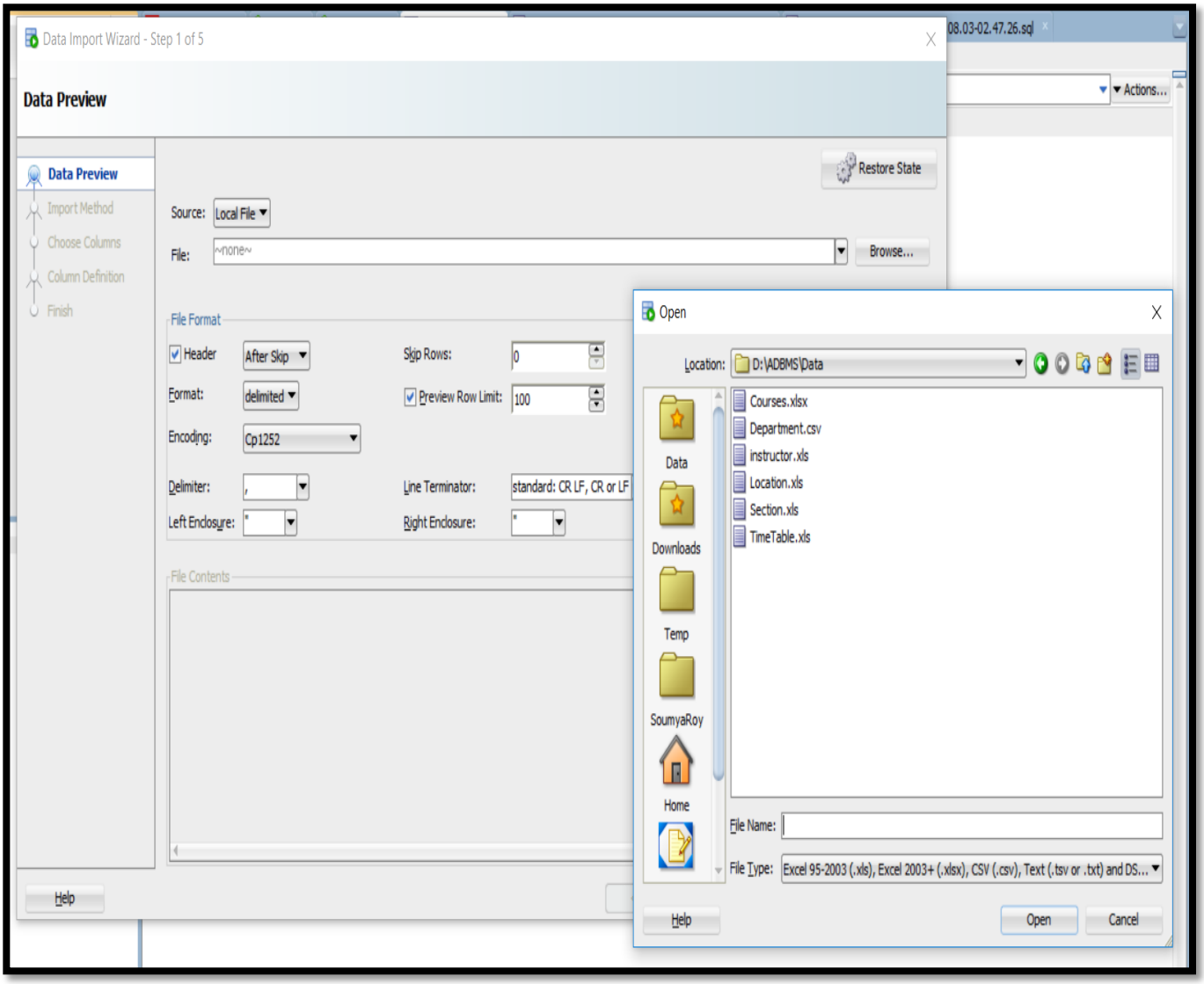
Data Insertion

Using Queries (Only a part is included in the screenshot)

Instructor Table:

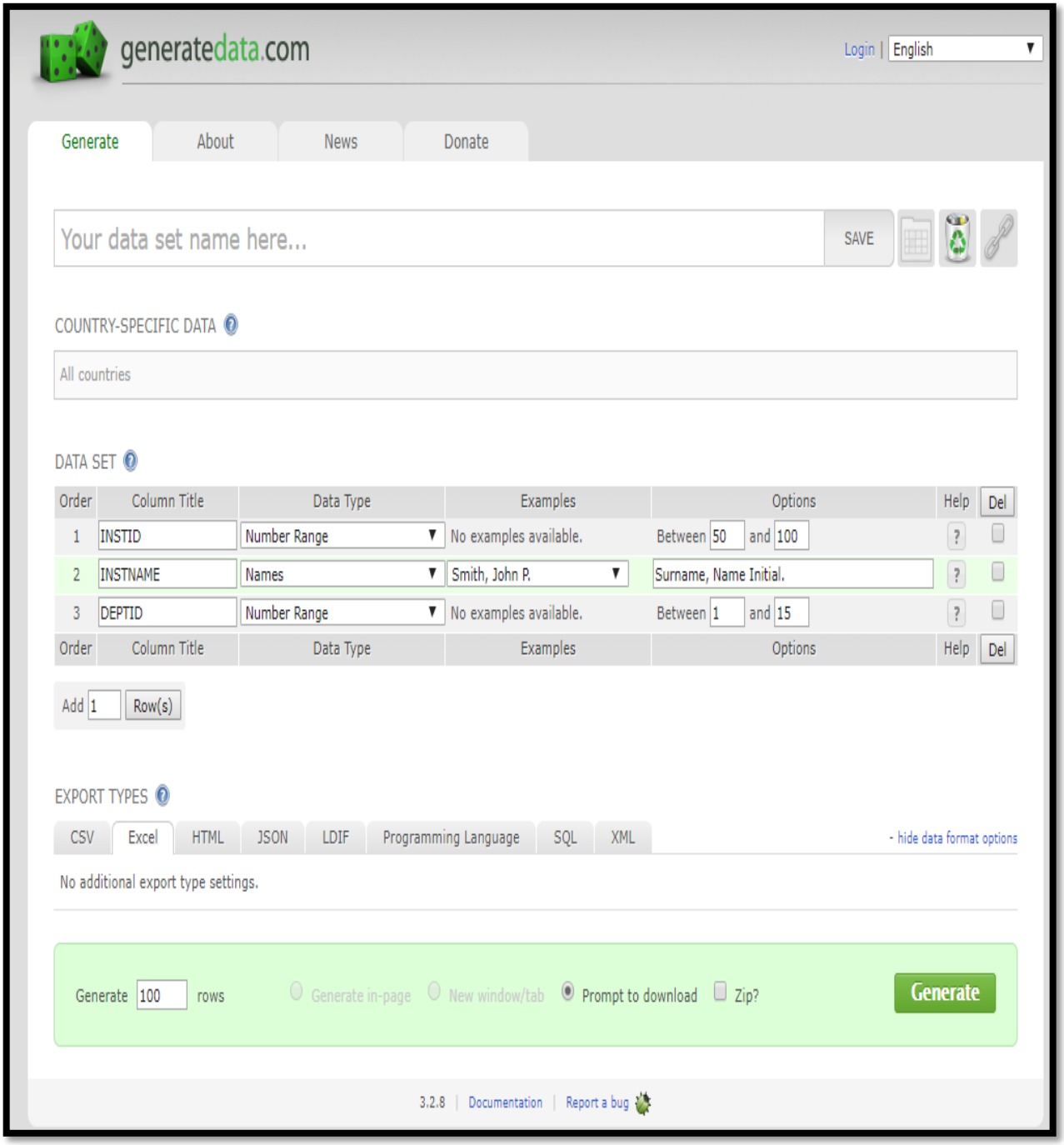
```
INSERT INTO `instructor` (`INSTID`,`INSTNAME`,`DEPTID`) VALUES (12,'Gavin Beasley',165);
INSERT INTO `instructor` (`INSTID`,`INSTNAME`,`DEPTID`) VALUES (49,'Samuel Guerrero',183);
INSERT INTO `instructor` (`INSTID`,`INSTNAME`,`DEPTID`) VALUES (41,'Duncan Bonner',173);
INSERT INTO `instructor` (`INSTID`,`INSTNAME`,`DEPTID`) VALUES (30,'Raphael Cooley',187);
INSERT INTO `instructor` (`INSTID`,`INSTNAME`,`DEPTID`) VALUES (37,'Harper Morse',184);
INSERT INTO `instructor` (`INSTID`,`INSTNAME`,`DEPTID`) VALUES (39,'Ginger Howe',172);
INSERT INTO `instructor` (`INSTID`,`INSTNAME`,`DEPTID`) VALUES (20,'Reese Brennan',183);
INSERT INTO `instructor` (`INSTID`,`INSTNAME`,`DEPTID`) VALUES (3,'Blaze Marshall',165);
INSERT INTO `instructor` (`INSTID`,`INSTNAME`,`DEPTID`) VALUES (20,'Hamish Fernandez',163);
INSERT INTO `instructor` (`INSTID`,`INSTNAME`,`DEPTID`) VALUES (14,'Leroy Watson',161);
INSERT INTO `instructor` (`INSTID`,`INSTNAME`,`DEPTID`) VALUES (41,'Hannah Raymond',200);
INSERT INTO `instructor` (`INSTID`,`INSTNAME`,`DEPTID`) VALUES (20,'Dale Rosario',153);
INSERT INTO `instructor` (`INSTID`,`INSTNAME`,`DEPTID`) VALUES (24,'Wayne Hopkins',187);
INSERT INTO `instructor` (`INSTID`,`INSTNAME`,`DEPTID`) VALUES (12,'Francesca Burks',197);
INSERT INTO `instructor` (`INSTID`,`INSTNAME`,`DEPTID`) VALUES (45,'Phyllis Ramos',195);
INSERT INTO `instructor` (`INSTID`,`INSTNAME`,`DEPTID`) VALUES (40,'Macaulay Fuentes',200);
INSERT INTO `instructor` (`INSTID`,`INSTNAME`,`DEPTID`) VALUES (34,'Myles Stafford',180);
INSERT INTO `instructor` (`INSTID`,`INSTNAME`,`DEPTID`) VALUES (26,'Bruce Valencia',181);
INSERT INTO `instructor` (`INSTID`,`INSTNAME`,`DEPTID`) VALUES (23,'Zachary Fulton',161);
INSERT INTO `instructor` (`INSTID`,`INSTNAME`,`DEPTID`) VALUES (41,'Colt Blair',166);
INSERT INTO `instructor` (`INSTID`,`INSTNAME`,`DEPTID`) VALUES (48,'Kim Golden',156);
INSERT INTO `instructor` (`INSTID`,`INSTNAME`,`DEPTID`) VALUES (30,'Jonah Chaney',161);
INSERT INTO `instructor` (`INSTID`,`INSTNAME`,`DEPTID`) VALUES (43,'Ivory Anderson',182);
INSERT INTO `instructor` (`INSTID`,`INSTNAME`,`DEPTID`) VALUES (3,'Kasper Faulkner',156);
INSERT INTO `instructor` (`INSTID`,`INSTNAME`,`DEPTID`) VALUES (14,'Gemma Paul',168);
INSERT INTO `instructor` (`INSTID`,`INSTNAME`,`DEPTID`) VALUES (43,'Cruz Patton',178);
INSERT INTO `instructor` (`INSTID`,`INSTNAME`,`DEPTID`) VALUES (9,'Fredericka Vargas',185);
INSERT INTO `instructor` (`INSTID`,`INSTNAME`,`DEPTID`) VALUES (23,'Jesse Perkins',191);
INSERT INTO `instructor` (`INSTID`,`INSTNAME`,`DEPTID`) VALUES (13,'Sybil Wong',168);
INSERT INTO `instructor` (`INSTID`,`INSTNAME`,`DEPTID`) VALUES (42,'Kitra Farley',176);
INSERT INTO `instructor` (`INSTID`,`INSTNAME`,`DEPTID`) VALUES (38,'Roth Kaufman',186);
INSERT INTO `instructor` (`INSTID`,`INSTNAME`,`DEPTID`) VALUES (13,'Russell Anderson',192);
INSERT INTO `instructor` (`INSTID`,`INSTNAME`,`DEPTID`) VALUES (44,'Malachi Beck',183);
INSERT INTO `instructor` (`INSTID`,`INSTNAME`,`DEPTID`) VALUES (1,'Kuame Gallegos',179);
INSERT INTO `instructor` (`INSTID`,`INSTNAME`,`DEPTID`) VALUES (33,'McKenzie Beard',170);
INSERT INTO `instructor` (`INSTID`,`INSTNAME`,`DEPTID`) VALUES (30,'Phyllis Figueroa',160);
INSERT INTO `instructor` (`INSTID`,`INSTNAME`,`DEPTID`) VALUES (28,'Cora Grant',177);
INSERT INTO `instructor` (`INSTID`,`INSTNAME`,`DEPTID`) VALUES (1,'Yen Farmer',153);
INSERT INTO `instructor` (`INSTID`,`INSTNAME`,`DEPTID`) VALUES (14,'Emmanuel Richmond',192);
INSERT INTO `instructor` (`INSTID`,`INSTNAME`,`DEPTID`) VALUES (35,'Jakeem Sosa',192);
INSERT INTO `instructor` (`INSTID`,`INSTNAME`,`DEPTID`) VALUES (47,'Xena Bender',159);
INSERT INTO `instructor` (`INSTID`,`INSTNAME`,`DEPTID`) VALUES (41,'Noel Mcknight',150);
INSERT INTO `instructor` (`INSTID`,`INSTNAME`,`DEPTID`) VALUES (38,'Hilda Bates',165);
```

Importing Data using csv/excel files



Data Generation

Using Online Data Generator:



Screenshots of Actual Data in Tables:

Instructor:

```
SELECT
    instid,
    instname,
    deptid
FROM
    instructor3
    ;
```

Script Output × | Query Result ×

SQL | Fetched 50 rows in 0.047 seconds

	INSTID	INSTNAME	DEPTID
1	64	Padmanabhan Balaji	13
2	69	Zantedeschi Daniel	6
3	56	Manish Agrawal	6
4	67	Dutta Kaushik	13
5	60	Alan Balfour	4
6	63	He Zhang	14
7	52	Clinton Daniel	5
8	45	Amena Goff	4
9	34	Slade Fernandez	11
10	9	Ivan Garner	2
11	16	Petra Bradford	5
12	21	Nola Moody	10
13	39	Ezekiel Lewis	1
14	22	Malik Woodard	11
15	28	Scott Hood	12
16	41	Gwendolyn Woodward	15
17	47	Heidi Gibbs	1

Location:

```
SELECT
    classroom,
    building,
    deptid
FROM
    location1;
```

Script Output × | Query Result ×

SQL | Fetched 50 rows in 0.051 seconds

	CLASSROOM	BUILDING	DEPTID
1	BSN 101	Muma Old building	7
2	CIS 201	Zimmerman	15
3	ESN 301	Engineering	14
4	B3R 4M7	Nexium	7
5	G1R 1B7	Sertraline HCl	12
6	Y5O 4J7	Hydrocodone/APAP	12
7	S0N 3N7	Fluticasone Propionate	4
8	A0V 3F4	TriNessa	11
9	P3O 1D6	Triamterene/Hydrochlorothiazide	8
10	U8C 9V7	Cyclobenzaprin HCl	8
11	W1E 6T6	LevothyroxineSodium	12
12	V9T 6T0	Namenda	3
13	H8G 0P1	Alprazolam	11
14	D9L 3Q1	Amoxicillin	7
15	J5V 7I8	Hydrochlorothiazide	3
16	I3C 0R8	Amoxicillin	9
17	A4X 6E8	Azithromycin	1

Section:

```sql
SELECT
    secid,
    term,
    startdate,
    enddate,
    day
FROM
    section1;
```

Script Output × Query Result ×

SQL | Fetched 50 rows in 0.062 seconds

	SECID	TERM	STARTDATE	ENDDATE	DAY
1	1	1	August 12th 2018	December 27th 2018	2
2	2	2	August 11th 2018	December 25th 2018	1
3	3	3	August 6th 2018	December 20th 2018	5
4	4	4	August 13th 2018	December 20th 2018	4
5	5	5	August 2nd 2018	December 16th 2018	3
6	6	6	August 5th 2018	December 18th 2018	1
7	7	7	August 9th 2018	December 17th 2018	2
8	8	8	August 12th 2018	December 17th 2018	2
9	9	9	August 10th 2018	December 22nd 2018	5
10	10	10	August 7th 2018	December 27th 2018	1
11	11	11	August 11th 2018	December 24th 2018	2
12	12	12	August 13th 2018	December 21st 2018	1
13	13	13	August 9th 2018	December 21st 2018	5
14	14	14	August 10th 2018	December 20th 2018	5

Department:

```
SELECT
    deptid,
    deptname
FROM
    department1;
```

Script Output × | Query Result ×

SQL | All Rows Fetched: 15 in 0.064 seconds

	DEPTID	DEPTNAME
1	1	Muma college of business
2	2	Zimmerman School of Advertising and Mass Communications
3	3	School of Engineering
4	4	Department of health science
5	5	Mathematics department
6	6	Chemistry department
7	7	Physics department
8	8	Morsani School of Medicine
9	9	Dentistry
10	10	School of Music
11	11	Department of Arts
12	12	Department of Social Sciences
13	13	School of Information Sciences
14	14	Marine Sciences

Courses:

```sql
SELECT
    courseid,
    coursename,
    instid,
    credits
FROM
    courses1;
```

Script Output | Query Result

SQL | Fetched 50 rows in 0.063 seconds

	COURSEID	COURSENAME	INSTID	CREDITS
1	6124	Adv system Analysis/Design	87	3
2	6136	Data Mining	81	3
3	6562	Big Data for Business	94	3
4	6137	Statistical data mining	52	3
5	6316	Project managemnet	93	3
6	6218	Advanced database management system	74	3
7	6021	Management information system	96	2
8	3232	Business application developement	59	3
9	3201	Accounting and Finance with European Study BSc	75	3
10	3154	Accounting and Finance with Industrial Experience BSc	54	3
11	3123	Accounting and Finance with International Study BSc	100	3
12	3648	Ancient History BA	65	3
13	3291	Data Science BSc	60	2

Timetable:

```
SELECT
    ttid,
    deptid,
    instid,
    courseid,
    secid
FROM
    timetable3;
```

Script Output × | Query Result ×

SQL | Fetched 50 rows in 0.05 seconds

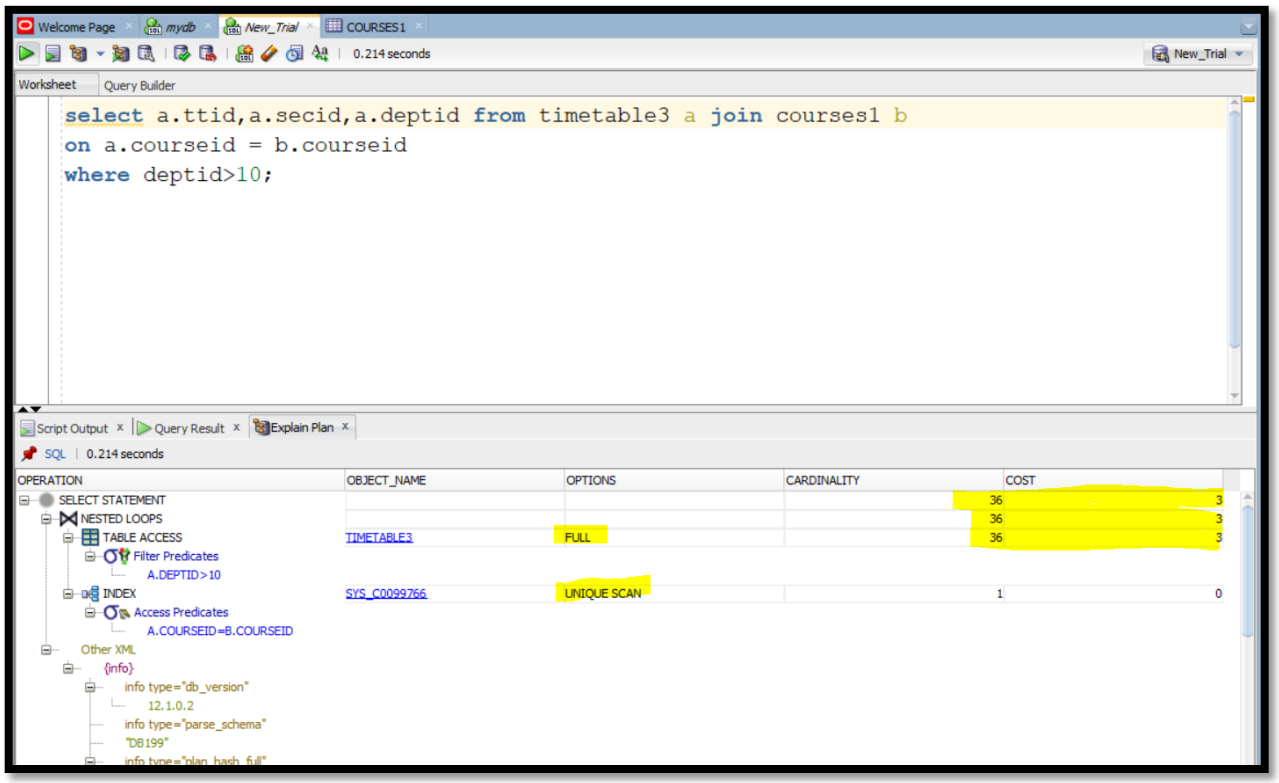
	TTID	DEPTID	INSTID	COURSEID	SECID
1	1	5	65	6124	36
2	2	15	91	6136	51
3	3	8	68	6562	74
4	4	11	90	6137	99
5	5	11	79	6316	76
6	6	8	98	6218	63
7	7	9	98	6021	92
8	8	13	90	3232	47
9	9	7	79	3201	8
10	10	3	56	3154	9
11	11	4	63	3123	18
12	12	15	82	3648	100
13	13	4	100	3291	69

Performance Tuning

Indexing

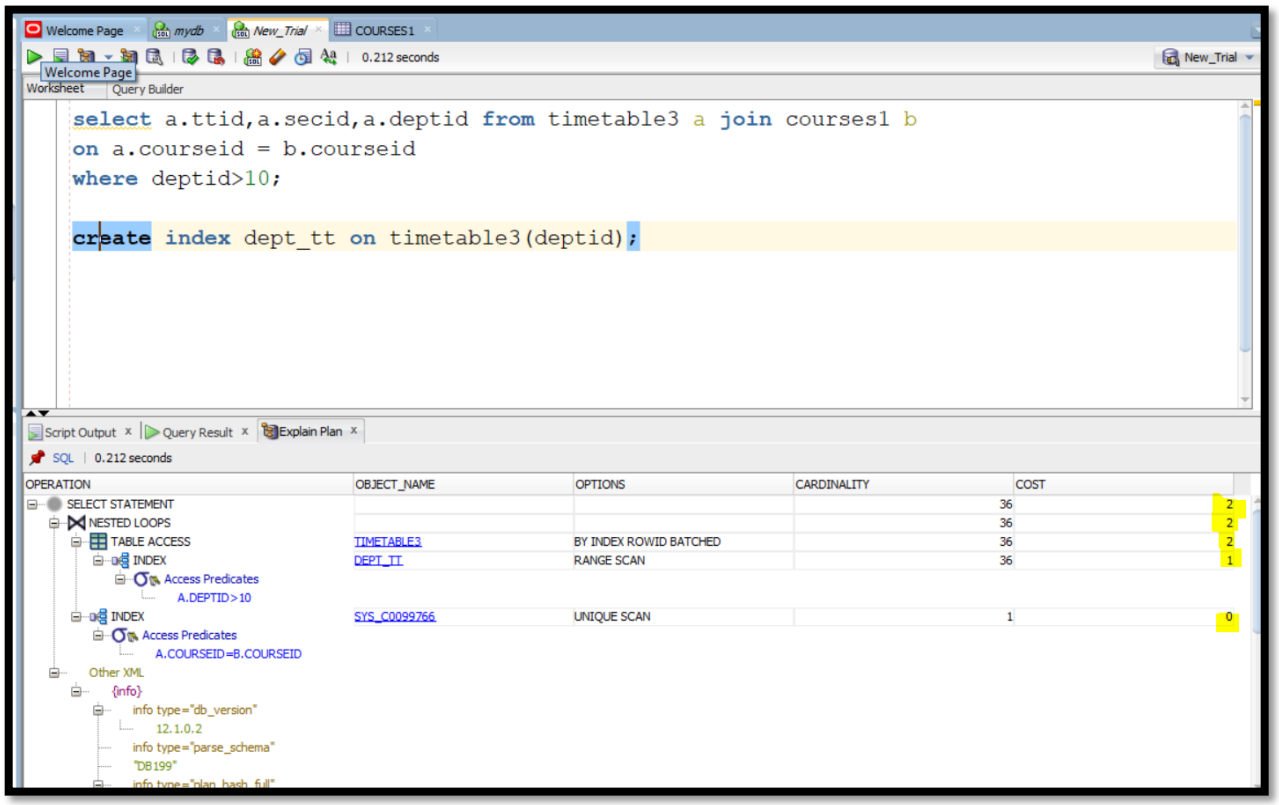
1. Querying to fetch the department IDS above 10.

   Let us look at the execution plan of the query



PS: Notice the highlighted parameters

Now, let us add an index to the ***DeptID*** column and see how it effects the execution plan of the query.



Things changed after indexing:

The Range Scan is included, and the cost reduces a lot.

Sample Queries

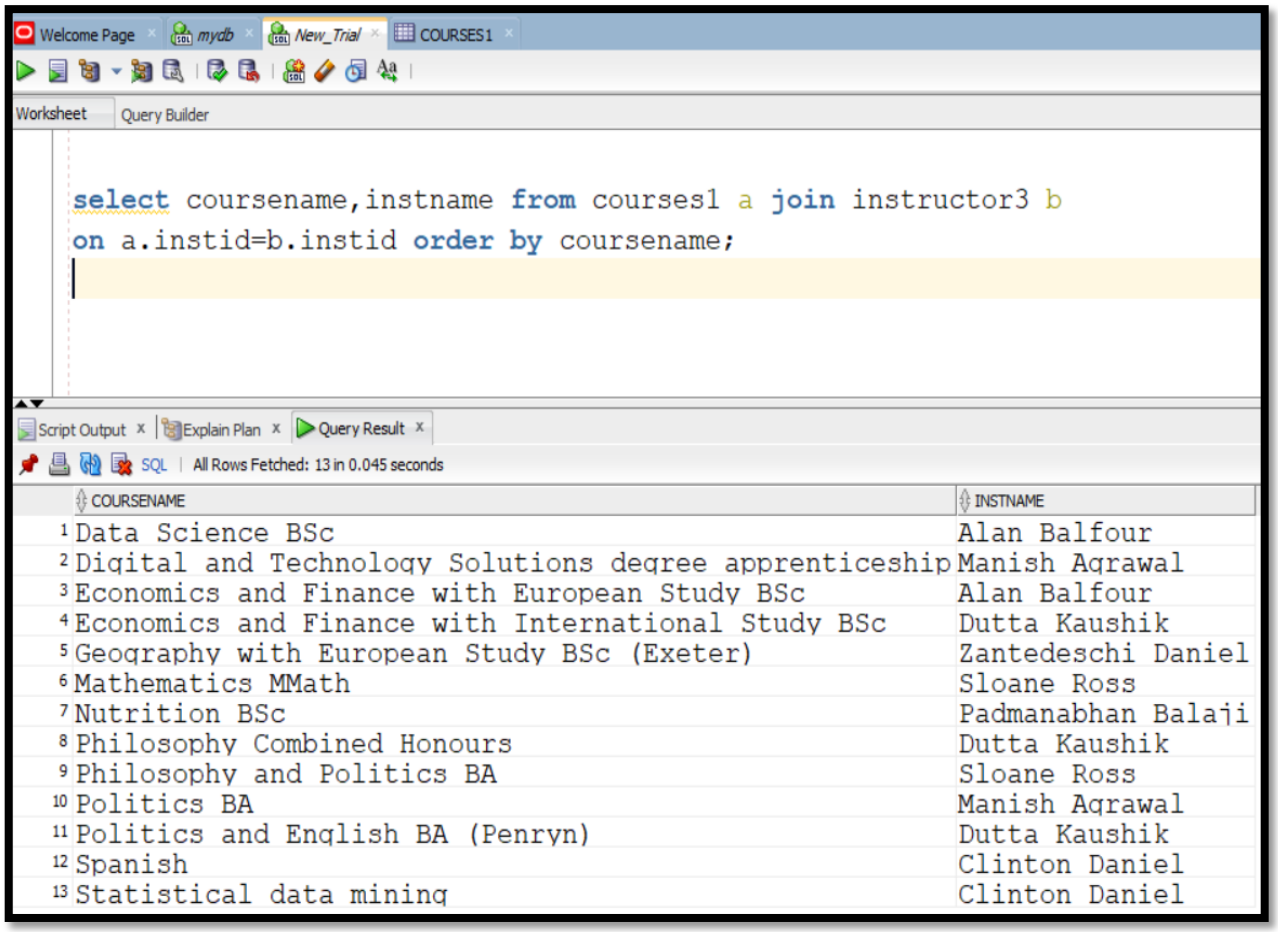
A) Find the classrooms of the buildings where the respective courses are taught in.

```
select coursename,z.classroom,z.building from courses1 x join
(select instid,instname,classroom,building,b.deptid from instructor3 a join location1 b
on a.deptid=b.deptid) z on x.instid=z.instid;
```

Fetched 50 rows in 0.048 seconds

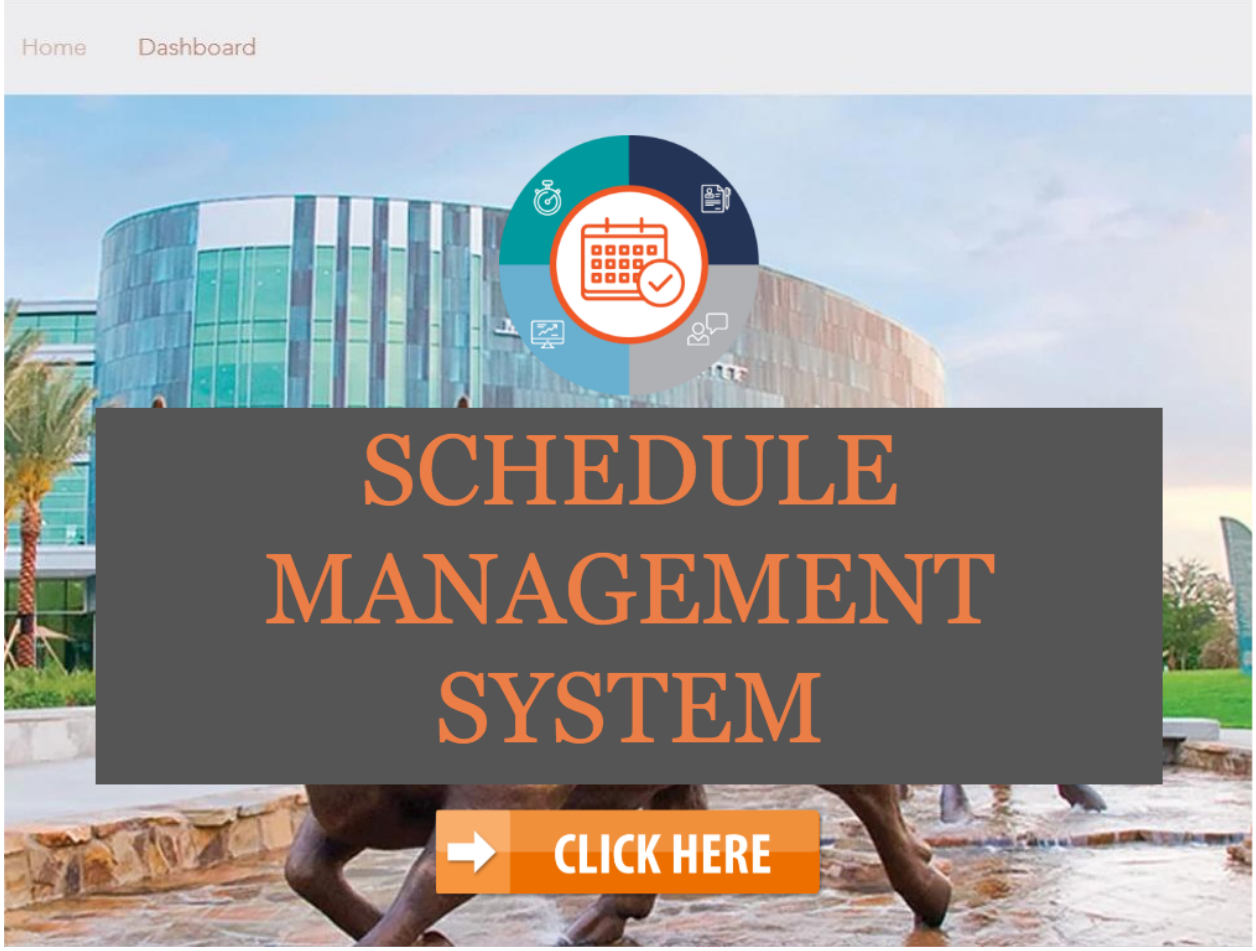
	COURSENAME	CLASSROOM	BUILDING
1	Data Science BSc	S0N 3N7	Fluticasone Propionate
2	Economics and Finance with European Study BSc	S0N 3N7	Fluticasone Propionate
3	Philosophy and Politics BA	X9W 2Q0	Metformin HCl
4	Mathematics MMath	X9W 2Q0	Metformin HCl
5	Data Science BSc	P9G 0S6	Alendronate Sodium
6	Economics and Finance with European Study BSc	P9G 0S6	Alendronate Sodium
7	Geography with European Study BSc (Exeter)	Z1H 9P2	Zolpidem Tartrate
8	Politics BA	Z1H 9P2	Zolpidem Tartrate
9	Digital and Technology Solutions degree apprenticeship	Z1H 9P2	Zolpidem Tartrate
10	Philosophy Combined Honours	K1D 0G6	Ibuprofen (Rx)
11	Politics and English BA (Penryn)	K1D 0G6	Ibuprofen (Rx)
12	Economics and Finance with International Study BSc	K1D 0G6	Ibuprofen (Rx)
13	Nutrition BSc	K1D 0G6	Ibuprofen (Rx)
14	Geography with European Study BSc (Exeter)	X4L 0U8	Lantus Solostar
15	Politics BA	X4L 0U8	Lantus Solostar
16	Digital and Technology Solutions degree apprenticeship	X4L 0U8	Lantus Solostar
17	Philosophy and Politics BA	N2E 2V9	Hydrocodone/APAP
18	Mathematics MMath	N2E 2V9	Hydrocodone/APAP

B) Find which instructor teaches what course?

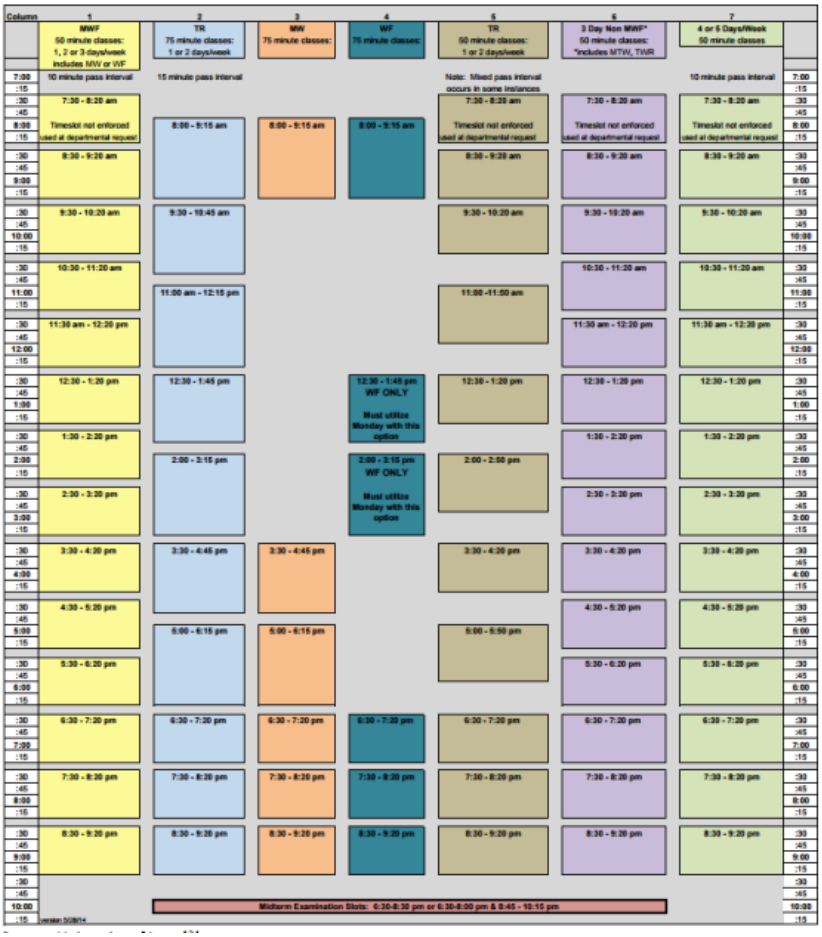


Data Visualization

We created a prototype that can be used in the future for the scheduler system. Below is its homepage



A similar dashboard can be used as a graphical user



Future Scope

1. This scheduler system can be extended to become part of a complete ERP solution that could run the entire subsystems at the university level
2. Using an agent enabled adaptive decision support system which employs software agents to assist humans in making strategic decisions under dynamic and uncertain conditions.