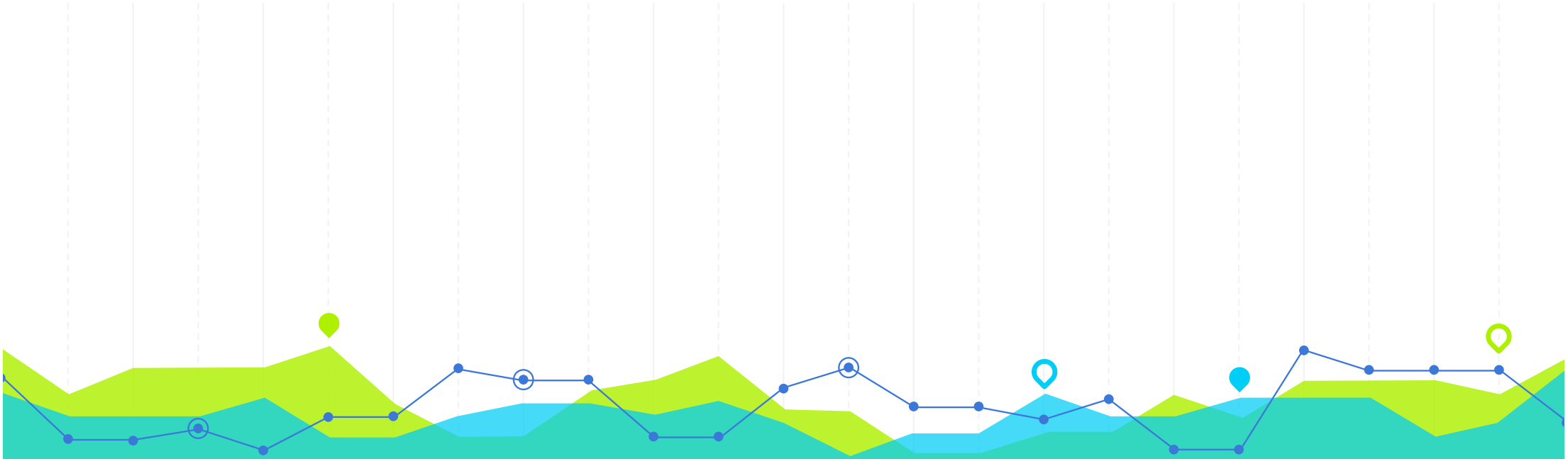
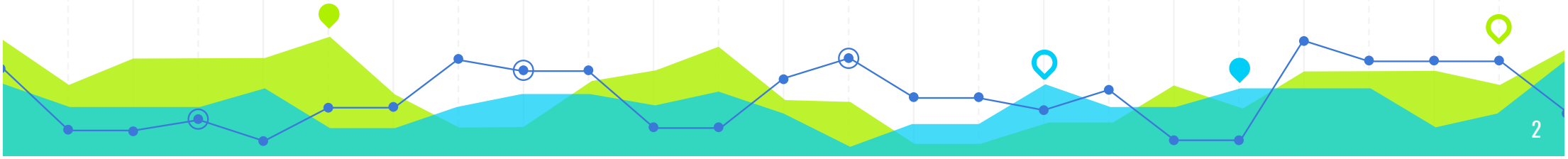


Lab 5: Nested Subqueries

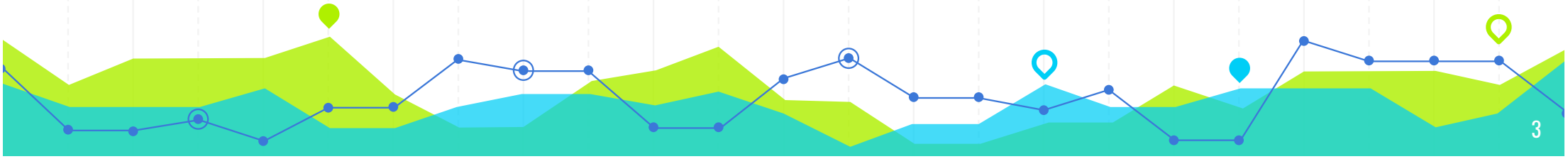
Set Membership

- SQL allows testing tuples for membership in a relation
- The **in** connective tests for set membership, where the set is a collection of values produced by a **select** clause
- The **not in** connective tests for the absence of set membership



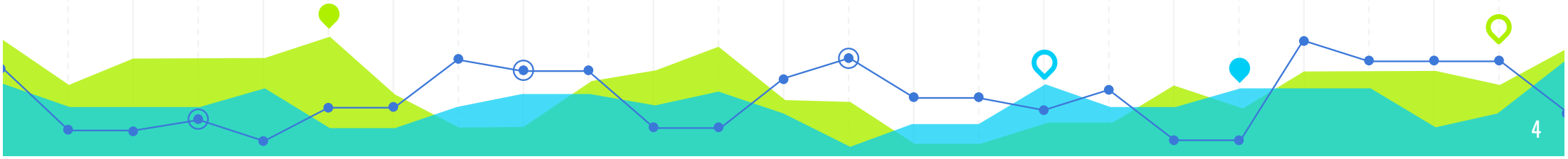
Set Membership

- Find all the courses taught in the both the Fall 2017 and Spring 2018 semesters.
  - **SELECT DISTINCT** *course_id*
    FROM *section*
    WHERE *semester = 'Fall'* **AND** *year= 2017* **AND**
    course_id **IN**
    (**SELECT** *course_id*
    FROM *section*
    WHERE *semester = 'Spring'* **AND** *year= 2018);*



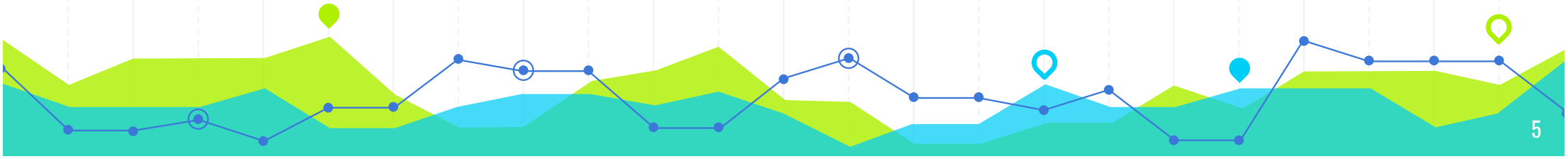
Set Membership

- Find all the courses taught in the Fall 2017 semester but not in the Spring 2018 semester.
  - **SELECT DISTINCT** *course_id*
    FROM *section*
    WHERE *semester = 'Fall'* **AND** *year= 2017* **AND**
    course_id **NOT IN**
    (**SELECT** *course_id*
    FROM *section*
    WHERE *semester = 'Spring'* **AND** *year= 2018);*



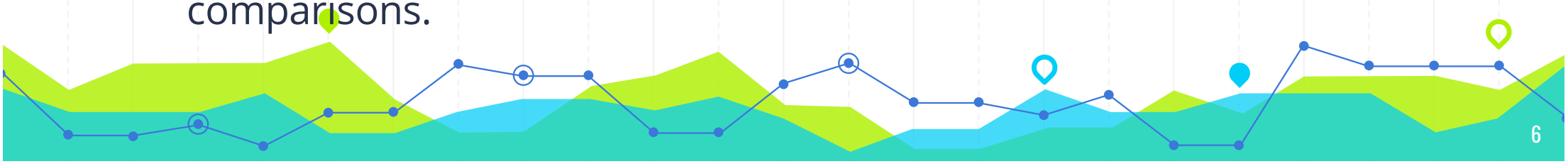
Set Membership

- Find the total number of (distinct) students who have taken course sections taught by the instructor with *ID 110011*
  - **SELECT COUNT** (**DISTINCT** ID*)*
    FROM *takes*
    WHERE (*course_id, sec_id, semester, year)* **IN**
    (**SELECT** *course_id, sec_id, semester, year*
    FROM *teaches*
    WHERE *teaches.ID= 10101);*



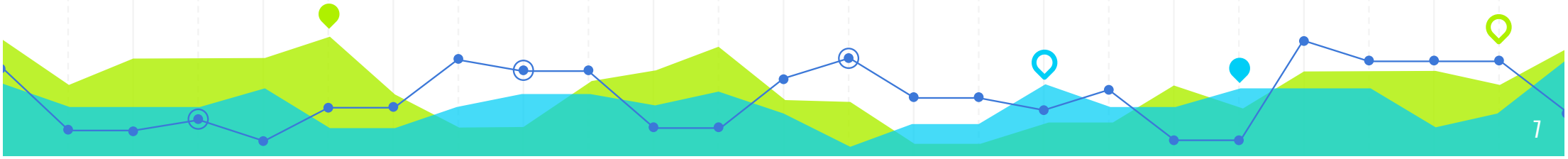
Set Comparison

- Find the names of all instructors whose salary is greater than at least one instructor in the Biology department.
  - **SELECT DISTINCT** *T.name*
    FROM *instructor* **AS** *T, instructor* **AS** *S*
    WHERE *T.salary* > *S.salary* **AND** *S.dept_name* = *'Biology';*
  - **SELECT** *name*
    FROM *instructor*
    WHERE *salary* > **SOME** (**SELECT** *salary*
    FROM *instructor* **WHERE** *dept_name* = *'Biology');*
- SQL also allows < **SOME**, <= **SOME**, >= **SOME**, = **SOME,** and <> **SOME** comparisons.



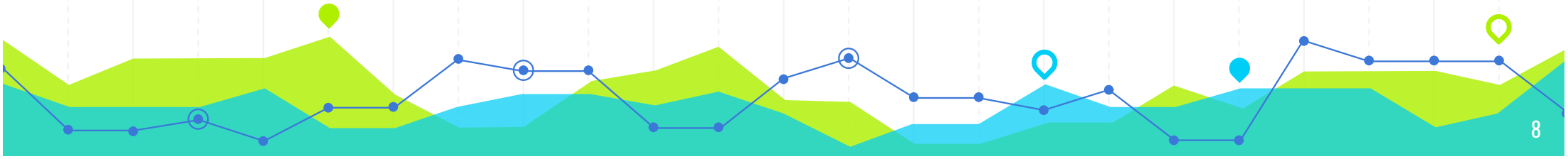
Set Comparison

- **SELECT** *name*
  FROM *instructor*
  WHERE *salary* > **ALL** (**SELECT** *salary*
  FROM *instructor* **WHERE** *dept_name = 'Biology');*
- SQL also allows < **ALL**, <= **ALL**, >= **ALL**, = **ALL** ,and <> **ALL** comparisons.



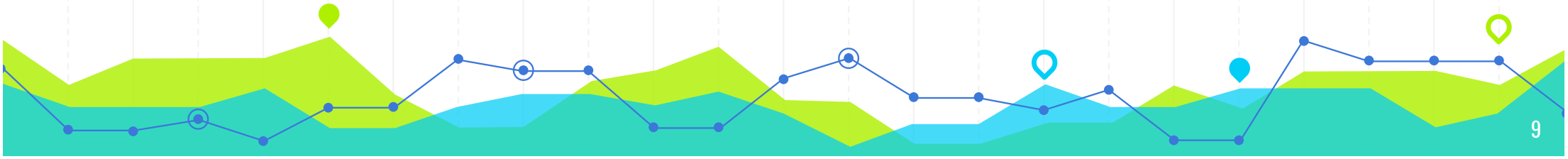
Set Comparison

- Find the departments that have the highest average salary.
  - **SELECT** *dept_name*
    FROM *instructor*
    GROUP BY *dept_name*
    HAVING AVG (*salary*) >= **ALL** (**SELECT AVG** (*salary*)
    FROM *instructor*
    GROUP BY *dept_name*);



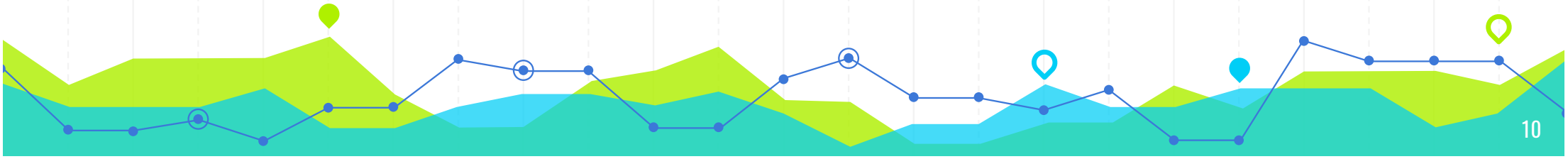
Test for Empty Relations

- Find all courses taught in both the Fall 2017 semester and in the Spring 2018 semester
  - **SELECT** *course_id*
    FROM *section* **AS** *S*
    WHERE *semester = 'Fall'* **AND** *year= 2017* **AND**
    EXISTS (**SELECT** * **FROM** *section* **AS** *T*
    WHERE *semester = 'Spring'* **AND** *year= 2018* **AND**
    S.course_id = T.course_id);



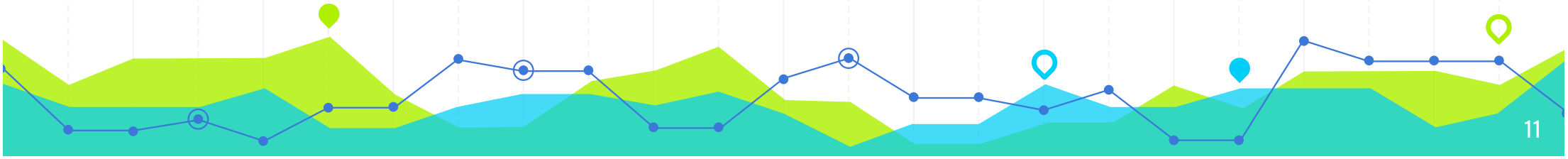
Test for the Absence of Duplicate Tuples

- Find all courses that were offered at most once in 2017
  - **SELECT** *T.course_id*
    FROM *course* **AS** *T*
    WHERE 1 <= (**SELECT COUNT**(*R.course_id)*
    FROM *section* **AS** *R*
    WHERE *T.course_id= R.course_id* **AND** *R.year = 2009);*



Homework

- Write the following queries in SQL, using the university schema.
  - Find the titles of courses in the Comp. Sci. department that have 3 credits.
  - Find the IDs of all students who were taught by an instructor named Einstein; make sure there are no duplicates in the result.
  - Find the highest salary of any instructor.
  - Find all instructors earning the highest salary (there may be more than one with the same salary).
  - Find the enrollment of each section that was offered in Autumn 2017.
  - Find the maximum enrollment, across all sections, in Autumn 2017.
  - Find the sections that had the maximum enrollment in Autumn 2017.



Homework

- Write the following queries in SQL, using the university schema.
  - Create a new course “CS-001”, titled “Weekly Seminar”, with 0 credits.
  - Create a section of this course in Autumn 2017, with *sec_id of 1.*
  - Enroll every student in the Comp. Sci. department in the above section.
  - Delete enrollments in the above section where the student’s name is Chavez.
  - Delete the course CS-001. What will happen if you run this delete statement without first deleting offerings (sections) of this course.
  - Delete all *takes tuples corresponding to any section of any course with* the word “database” as a part of the title; ignore case when matching the word with the title