

1.

**{EmpID}**: Each employee has a unique EmpID.

**{SSN}**: Social Security Numbers (SSN) are unique identifiers.

**{Email}**: Each employee has a unique email address.

**{EmpID, SSN}**: Combining these two unique attributes still forms a superkey.

**{EmpID, Email}**: Another combination of unique attributes.

**{EmpID, Phone, SSN}**: Even though EmpID alone is a superkey, adding other attributes still makes it one.

A **superkey** is a set of one or more attributes that, taken together, can uniquely identify a row in a table.

2.

**{EmpID}**: It's a superkey, and removing EmpID would not leave a unique identifier.

**{SSN}**: It's a superkey, and removing SSN would not leave a unique identifier.

**{Email}**: It's a superkey, and removing Email would not leave a unique identifier.

A **candidate key** is a **minimal** superkey, meaning you can't remove any attribute from it and still maintain uniqueness. It's a key that can be chosen as the primary key.

**3. Primary Key Selection**

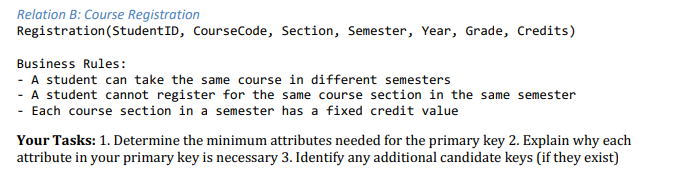
For a **primary key**, you'd typically choose **{EmpID}**.

A good primary key should be simple, stable, and have no business meaning that might change.

**4. Can Two Employees Have the Same Phone Number?**

Yes, two employees **can** have the same phone number based on the given data.

For example, a shared family phone line. Therefore, Phone is not a candidate key.



**1. Primary Key Analysis**

The minimum attributes needed for the primary key are **{StudentID, CourseCode, Section, Semester, Year}**.

**2. Explanation of Each Attribute's Necessity**

**StudentID**: This is a fundamental part of the key. Without it, you couldn't distinguish between two different students taking the same course.

**CourseCode**: This is also essential. A single student can register for multiple courses, so you need this attribute to differentiate between them.

**Section**: According to the business rules, a student cannot register for the same course section in the same semester. This implies that the Section is needed to uniquely identify the course instance within a semester, in case there are multiple sections of the same course.

**Semester** and **Year**: The first business rule states that "A student can take the same course in different semesters." This means that Semester and Year are required to differentiate between a student's registration for the same course across different academic terms. For example, a student might take 'MATH101' in Fall 2024 and then retake it in Spring 2025.

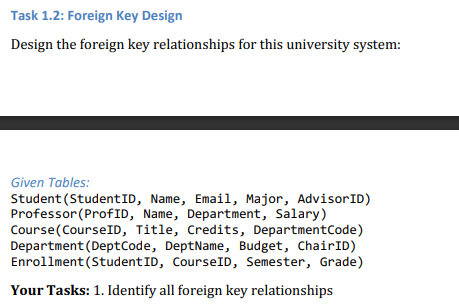
**3. Additional Candidate Keys**

Based on the provided information, there are **no additional candidate keys**.

A candidate key must be a minimal superkey, meaning all its attributes are necessary for unique identification.

The attributes Grade and Credits cannot be part of a candidate key because they are dependent on the other attributes and do not uniquely identify a row. A student can take the same course multiple times, and their grade or credits could be different for each attempt.

The combination of **{StudentID, CourseCode, Section, Semester, Year}** is the only set of attributes that guarantees uniqueness based on the provided business rules. Any smaller combination, as explained above, would fail to uniquely identify a record.



1. **Student Table**:

**AdvisorID** in the Student table is a foreign key that references **ProfID** (the primary key) in the Professor table. This links each student to their academic advisor.

1. **Course Table**:

**DepartmentCode** in the Course table is a foreign key that references **DeptCode** (the primary key) in the Department table. This connects each course to its respective department.

1. **Department Table**:

**ChairID** in the Department table is a foreign key that references **ProfID** (the primary key) in the Professor table. This identifies which professor is the head of the department.

1. **Enrollment Table**:

**StudentID** in the Enrollment table is a foreign key that references **StudentID** (the primary key) in the Student table. This links an enrollment record to a specific student.

**CourseID** in the Enrollment table is a foreign key that references **CourseID** (the primary key) in the Course table. This links an enrollment record to a specific course.