**Question**: What are the first names of all employees? **SQL Query**:

sql

Copy code

SELECT first\_name FROM Employees;

**Question**: List the last names and ages of employees older than 30. **SQL Query**:

sql

Copy code

SELECT last\_name, age FROM Employees WHERE age > 30;

**Question**: Show the full details of employees who earn more than 60000. **SQL Query**:

sql

Copy code

SELECT \* FROM Employees WHERE salary > 60000;

**Question**: What are the names and salaries of employees whose last name is 'Mukherjee'? **SQL Query**:

sql

Copy code

SELECT first\_name, last\_name, salary FROM Employees WHERE last\_name = 'Mukherjee';

**Question**: How many employees are there in the database? **SQL Query**:

sql

Copy code

SELECT COUNT(\*) FROM Employees;

**Question**: List the full names of employees sorted by age in descending order. **SQL Query**:

sql

Copy code

SELECT first\_name, last\_name FROM Employees ORDER BY age DESC;

**Question**: What is the average salary of employees? **SQL Query**:

sql

Copy code

SELECT AVG(salary) FROM Employees;

**Question**: Show the full details of the youngest employee. **SQL Query**:

sql

Copy code

SELECT \* FROM Employees ORDER BY age ASC LIMIT 1;

**Question**: What is the total salary expenditure for all employees? **SQL Query**:

sql

Copy code

SELECT SUM(salary) FROM Employees;

**Question**: List the IDs and names of employees aged between 25 and 35. **SQL Query**:

sql

Copy code

SELECT id\_no, first\_name, last\_name FROM Employees WHERE age BETWEEN 25 AND 35;

**Question**: What are the names of employees who are exactly 30 years old? **SQL Query**:

sql

Copy code

SELECT first\_name, last\_name FROM Employees WHERE age = 30;

**Question**: List the employees' first names and their salaries, but only show those with a salary less than 60000. **SQL Query**:

sql

Copy code

SELECT first\_name, salary FROM Employees WHERE salary < 60000;

**Question**: What is the highest salary in the employee database? **SQL Query**:

sql

Copy code

SELECT MAX(salary) FROM Employees;

**Question**: Show the full details of employees with the last name starting with 'S'. **SQL Query**:

sql

Copy code

SELECT \* FROM Employees WHERE last\_name LIKE 'S%';

**Question**: List the employees whose first name contains 'ai'. **SQL Query**:

sql

Copy code

SELECT \* FROM Employees WHERE first\_name LIKE '%ai%';

These questions and SQL queries cover a range of typical queries that might be made on an employee database.

4o

ALL the synonyms and with stopword removed

('Find', {'rule', 'chance', 'ascertain', 'regain', 'see', 'breakthrough', 'feel', 'observe', 'encounter', 'come\_up', 'get', 'obtain', 'recover', 'detect', 'find\_out', 'witness', 'notice', 'receive', 'discovery', 'happen', 'uncovering', 'determine', 'line\_up', 'get\_hold', 'retrieve', 'discover', 'incur', 'find\_oneself', 'find', 'bump'}), ('employee', {'employee'}), ('age', {'maturate', 'eld', 'old\_age', 'geezerhood', 'mature', 'long\_time', 'age', 'historic\_period', 'get\_on', 'years', 'senesce'}), ('30', {'30', 'XXX', 'xxx', 'thirty'}), ('high', {'eminent', 'high\_gear', 'in\_high\_spirits', 'high\_up', 'highschool', 'high', 'gamy', 'high-pitched', 'senior\_high', 'luxuriously', 'richly', 'heights', 'gamey', 'high\_school', 'senior\_high\_school', 'mellow'}), ('salary', {'remuneration', 'wage', 'salary', 'pay', 'earnings'}), ('.', set())]

[('Show', {'testify', 'display', 'evidence', 'render', 'exhibit', 'evince', 'point', 'appearance', 'demonstrate', 'bear\_witness', 'designate', 'present', 'express', 'indicate', 'register', 'picture', 'shew', 'show\_up', 'show', 'demo', 'read', 'record', 'prove', 'establish', 'usher', 'depict'}), ('employee', {'employee'}), ('old', {'Old', 'erstwhile', 'one-time', 'honest-to-goodness', 'sure-enough', 'previous', 'old', 'quondam', 'older', 'honest-to-god', 'sometime', 'onetime', 'former'}), ('25', {'xxv', 'XXV', 'twenty-five', '25'}), ('.', set())]

[('List', {'heel', 'name', 'leaning', 'inclination', 'tilt', 'list', 'number', 'lean', 'listing'}), ('employee', {'employee'}), ('low', {'small', 'first\_gear', 'Sir\_David\_Alexander\_Cecil\_Low', 'depression', 'low-spirited', 'down\_in\_the\_mouth', 'miserable', 'low-toned', 'David\_Low', 'first', 'downhearted', 'humiliated', 'crushed', 'moo', 'modest', 'Sir\_David\_Low', 'blue', 'abject', 'depleted', 'low', 'Low', 'low\_gear', 'scummy', 'dispirited', 'lowly', 'depressed', 'broken', 'grim', 'low-down', 'downcast', 'low-pitched', 'gloomy', 'humbled', 'down', 'scurvy', 'humble'}), ('salary', {'remuneration', 'wage', 'salary', 'pay', 'earnings'}), ('.', set())]

[('Get', {'capture', 'pay\_back', 'develop', 'experience', 'bugger\_off', 'drive', 'aim', 'arrest', 'make', 'pose', 'flummox', 'catch', 'contract', 'fuck\_off', "get\_under\_one's\_skin", 'convey', 'stupefy', 'baffle', 'bring\_forth', 'start', 'father', 'set\_about', 'cause', 'start\_out', 'have', 'dumbfound', 'generate', 'acquire', 'vex', 'induce', 'scram', 'begin', 'fetch', 'sire', 'stick', 'sustain', 'get', 'gravel', 'suffer', 'obtain', 'take', 'bring', 'puzzle', 'bewilder', 'come', 'perplex', 'set\_out', 'mother', 'fix', 'produce', 'amaze', 'buzz\_off', 'receive', 'let', 'beget', 'nonplus', 'mystify', 'stimulate', 'engender', 'draw', 'go', 'commence', 'incur', 'pay\_off', 'grow', 'arrive', 'find', 'get\_down', 'become', 'beat'}), ('detail', {'point', 'item', 'detail', 'particular', 'contingent'}), ('employee', {'employee'}), ('salary', {'remuneration', 'wage', 'salary', 'pay', 'earnings'}), ('50000', set()), ('.', set())]

[('first', {'kickoff', 'firstly', '1st', 'showtime', 'first\_gear', 'world-class', 'first\_base', 'start', 'outset', 'first', 'commencement', 'number\_one', 'initiative', 'first\_off', 'first-class\_honours\_degree', 'inaugural', 'starting\_time', 'low', 'number\_1', 'low\_gear', 'foremost', 'beginning', 'initiatory', 'maiden', 'first\_of\_all', 'offset', 'get-go', 'for\_the\_first\_time'}), ('name', {'epithet', 'make', 'public\_figure', 'diagnose', 'identify', 'figure', 'advert', 'gens', 'appoint', 'constitute', 'name', 'key', 'bring\_up', 'key\_out', 'describe', 'mention', 'nominate', 'cite', 'call', 'list', 'discover', 'distinguish', 'refer'}), ('employee', {'employee'}), ('?', set())]

[('List', {'heel', 'name', 'leaning', 'inclination', 'tilt', 'list', 'number', 'lean', 'listing'}), ('last', {'finally', 'survive', 'in\_conclusion', 'lowest', 'last-place', "shoemaker's\_last", 'net', 'finis', 'live\_on', 'terminal', 'final\_stage', 'endure', 'death', 'end', 'last', 'hold\_up', 'finish', 'lastly', 'close', 'stopping\_point', 'final', 'utmost', 'live', 'finale', "cobbler's\_last", 'go', 'concluding', 'conclusion', 'hold\_out'}), ('name', {'epithet', 'make', 'public\_figure', 'diagnose', 'identify', 'figure', 'advert', 'gens', 'appoint', 'constitute', 'name', 'key', 'bring\_up', 'key\_out', 'describe', 'mention', 'nominate', 'cite', 'call', 'list', 'discover', 'distinguish', 'refer'}), ('age', {'maturate', 'eld', 'old\_age', 'geezerhood', 'mature', 'long\_time', 'age', 'historic\_period', 'get\_on', 'years', 'senesce'}), ('employee', {'employee'}), ('old', {'Old', 'erstwhile', 'one-time', 'honest-to-goodness', 'sure-enough', 'previous', 'old', 'quondam', 'older', 'honest-to-god', 'sometime', 'onetime', 'former'}), ('30', {'30', 'XXX', 'xxx', 'thirty'}), ('.', set())]

[('Show', {'testify', 'display', 'evidence', 'render', 'exhibit', 'evince', 'point', 'appearance', 'demonstrate', 'bear\_witness', 'designate', 'present', 'express', 'indicate', 'register', 'picture', 'shew', 'show\_up', 'show', 'demo', 'read', 'record', 'prove', 'establish', 'usher', 'depict'}), ('full', {'replete', 'entire', 'fully', 'full\_moon', 'broad', 'full\_phase\_of\_the\_moon', 'wide-cut', 'to\_the\_full', 'total', 'wide', 'wax', 'full', 'full-of-the-moon', 'good'}), ('detail', {'point', 'item', 'detail', 'particular', 'contingent'}), ('employee', {'employee'}), ('earn', {'realise', 'clear', 'gain', 'bring\_in', 'make', 'realize', 'take\_in', 'garner', 'pull\_in', 'earn'}), ('60000', set()), ('.', set())]

[('name', {'epithet', 'make', 'public\_figure', 'diagnose', 'identify', 'figure', 'advert', 'gens', 'appoint', 'constitute', 'name', 'key', 'bring\_up', 'key\_out', 'describe', 'mention', 'nominate', 'cite', 'call', 'list', 'discover', 'distinguish', 'refer'}), ('salary', {'remuneration', 'wage', 'salary', 'pay', 'earnings'}), ('employee', {'employee'}), ('whose', set()), ('last', {'finally', 'survive', 'in\_conclusion', 'lowest', 'last-place', "shoemaker's\_last", 'net', 'finis', 'live\_on', 'terminal', 'final\_stage', 'endure', 'death', 'end', 'last', 'hold\_up', 'finish', 'lastly', 'close', 'stopping\_point', 'final', 'utmost', 'live', 'finale', "cobbler's\_last", 'go', 'concluding', 'conclusion', 'hold\_out'}), ('name', {'epithet', 'make', 'public\_figure', 'diagnose', 'identify', 'figure', 'advert', 'gens', 'appoint', 'constitute', 'name', 'key', 'bring\_up', 'key\_out', 'describe', 'mention', 'nominate', 'cite', 'call', 'list', 'discover', 'distinguish', 'refer'}), ('"', set()), ('Mukherjee', set()), ('"?', set())]

[('many', {'many'}), ('employee', {'employee'}), ('database', {'database'}), ('?', set()), ('List', {'heel', 'name', 'leaning', 'inclination', 'tilt', 'list', 'number', 'lean', 'listing'}), ('full', {'replete', 'entire', 'fully', 'full\_moon', 'broad', 'full\_phase\_of\_the\_moon', 'wide-cut', 'to\_the\_full', 'total', 'wide', 'wax', 'full', 'full-of-the-moon', 'good'}), ('name', {'epithet', 'make', 'public\_figure', 'diagnose', 'identify', 'figure', 'advert', 'gens', 'appoint', 'constitute', 'name', 'key', 'bring\_up', 'key\_out', 'describe', 'mention', 'nominate', 'cite', 'call', 'list', 'discover', 'distinguish', 'refer'}), ('employee', {'employee'}), ('sort', {'sort', 'sort\_out', 'kind', 'variety', 'screen', 'form', 'sieve', 'separate', 'class', 'sorting', 'classify', 'screen\_out', 'assort'}), ('age', {'maturate', 'eld', 'old\_age', 'geezerhood', 'mature', 'long\_time', 'age', 'historic\_period', 'get\_on', 'years', 'senesce'}), ('descend', {'go\_down', 'fall', 'come\_down', 'deign', 'descend', 'derive', 'condescend', 'come', 'settle'}), ('order', {'rules\_of\_order', 'say', 'rate', 'enjoin', 'orderliness', 'ordain', 'put', 'gild', 'Order', 'govern', 'guild', 'lodge', 'regularise', 'tell', 'fiat', 'order', 'prescribe', 'grade', 'decree', 'club', 'monastic\_order', 'edict', 'rescript', 'order\_of\_magnitude', 'parliamentary\_procedure', 'rank', 'dictate', 'Holy\_Order', 'purchase\_order', 'social\_club', 'society', 'ordering', 'ordinate', 'place', 'ordination', 'regulate', 'regularize', 'arrange', 'consecrate', 'range', 'parliamentary\_law', 'set\_up'}), ('.', set())]

[('average', {'average\_out', 'average', 'mean', 'medium', 'modal', 'median', 'fair', 'norm', 'intermediate', 'middling', 'mediocre', 'ordinary'}), ('salary', {'remuneration', 'wage', 'salary', 'pay', 'earnings'}), ('employee', {'employee'}), ('?', set())]

[('Show', {'testify', 'display', 'evidence', 'render', 'exhibit', 'evince', 'point', 'appearance', 'demonstrate', 'bear\_witness', 'designate', 'present', 'express', 'indicate', 'register', 'picture', 'shew', 'show\_up', 'show', 'demo', 'read', 'record', 'prove', 'establish', 'usher', 'depict'}), ('full', {'replete', 'entire', 'fully', 'full\_moon', 'broad', 'full\_phase\_of\_the\_moon', 'wide-cut', 'to\_the\_full', 'total', 'wide', 'wax', 'full', 'full-of-the-moon', 'good'}), ('detail', {'point', 'item', 'detail', 'particular', 'contingent'}), ('young', {'Lester\_Willis\_Young', 'youth', 'Pres\_Young', 'Whitney\_Moore\_Young\_Jr.', 'Cy\_Young', 'Edward\_Young', 'Brigham\_Young', 'new', 'Thomas\_Young', 'vernal', 'Whitney\_Young', 'Loretta\_Young', 'untested', 'offspring', 'young', 'Danton\_True\_Young', 'Young', 'youthful', 'unseasoned', 'immature', 'untried'}), ('employee', {'employee'}), ('.', set())]

[('total', {'add', 'tot', 'sum\_up', 'totality', 'add\_together', 'tally', 'entire', 'summate', 'sum', 'tot\_up', 'number', 'come', 'amount', 'total', 'aggregate', 'full', 'tote\_up', 'add\_up'}), ('salary', {'remuneration', 'wage', 'salary', 'pay', 'earnings'}), ('expenditure', {'using\_up', 'expenditure', 'outgo', 'expending', 'spending', 'consumption', 'outlay'}), ('employee', {'employee'}), ('?', set())]

[('List', {'heel', 'name', 'leaning', 'inclination', 'tilt', 'list', 'number', 'lean', 'listing'}), ('IDs', {'ID', 'id', 'I.D.', 'Gem\_State', 'Idaho'}), ('name', {'epithet', 'make', 'public\_figure', 'diagnose', 'identify', 'figure', 'advert', 'gens', 'appoint', 'constitute', 'name', 'key', 'bring\_up', 'key\_out', 'describe', 'mention', 'nominate', 'cite', 'call', 'list', 'discover', 'distinguish', 'refer'}), ('employee', {'employee'}), ('age', {'maturate', 'eld', 'old\_age', 'geezerhood', 'mature', 'long\_time', 'age', 'historic\_period', 'get\_on', 'years', 'senesce'}), ('25', {'xxv', 'XXV', 'twenty-five', '25'}), ('35', {'35', 'xxxv', 'thirty-five'}), ('.', set())]

[('name', {'epithet', 'make', 'public\_figure', 'diagnose', 'identify', 'figure', 'advert', 'gens', 'appoint', 'constitute', 'name', 'key', 'bring\_up', 'key\_out', 'describe', 'mention', 'nominate', 'cite', 'call', 'list', 'discover', 'distinguish', 'refer'}), ('employee', {'employee'}), ('exactly', {'on\_the\_dot', 'on\_the\_nose', 'incisively', 'exactly', 'just', 'precisely', 'on\_the\_button'}), ('30', {'30', 'XXX', 'xxx', 'thirty'}), ('year', {'year', 'yr', 'twelvemonth', 'class'}), ('old', {'Old', 'erstwhile', 'one-time', 'honest-to-goodness', 'sure-enough', 'previous', 'old', 'quondam', 'older', 'honest-to-god', 'sometime', 'onetime', 'former'}), ('?', set())]

[('List', {'heel', 'name', 'leaning', 'inclination', 'tilt', 'list', 'number', 'lean', 'listing'}), ('employee', {'employee'}), ("'", set()), ('first', {'kickoff', 'firstly', '1st', 'showtime', 'first\_gear', 'world-class', 'first\_base', 'start', 'outset', 'first', 'commencement', 'number\_one', 'initiative', 'first\_off', 'first-class\_honours\_degree', 'inaugural', 'starting\_time', 'low', 'number\_1', 'low\_gear', 'foremost', 'beginning', 'initiatory', 'maiden', 'first\_of\_all', 'offset', 'get-go', 'for\_the\_first\_time'}), ('name', {'epithet', 'make', 'public\_figure', 'diagnose', 'identify', 'figure', 'advert', 'gens', 'appoint', 'constitute', 'name', 'key', 'bring\_up', 'key\_out', 'describe', 'mention', 'nominate', 'cite', 'call', 'list', 'discover', 'distinguish', 'refer'}), ('salary', {'remuneration', 'wage', 'salary', 'pay', 'earnings'}), (',', set()), ('show', {'testify', 'display', 'evidence', 'render', 'exhibit', 'evince', 'point', 'appearance', 'demonstrate', 'bear\_witness', 'designate', 'present', 'express', 'indicate', 'register', 'picture', 'shew', 'show\_up', 'show', 'demo', 'read', 'record', 'prove', 'establish', 'usher', 'depict'}), ('salary', {'remuneration', 'wage', 'salary', 'pay', 'earnings'}), ('less', {'lupus\_erythematosus', 'LE', 'less', 'to\_a\_lesser\_extent'}), ('60000', set()), ('.', set())]

[('high', {'eminent', 'high\_gear', 'in\_high\_spirits', 'high\_up', 'highschool', 'high', 'gamy', 'high-pitched', 'senior\_high', 'luxuriously', 'richly', 'heights', 'gamey', 'high\_school', 'senior\_high\_school', 'mellow'}), ('salary', {'remuneration', 'wage', 'salary', 'pay', 'earnings'}), ('employee', {'employee'}), ('database', {'database'}), ('?', set())]

[('Show', {'testify', 'display', 'evidence', 'render', 'exhibit', 'evince', 'point', 'appearance', 'demonstrate', 'bear\_witness', 'designate', 'present', 'express', 'indicate', 'register', 'picture', 'shew', 'show\_up', 'show', 'demo', 'read', 'record', 'prove', 'establish', 'usher', 'depict'}), ('full', {'replete', 'entire', 'fully', 'full\_moon', 'broad', 'full\_phase\_of\_the\_moon', 'wide-cut', 'to\_the\_full', 'total', 'wide', 'wax', 'full', 'full-of-the-moon', 'good'}), ('detail', {'point', 'item', 'detail', 'particular', 'contingent'}), ('employee', {'employee'}), ('last', {'finally', 'survive', 'in\_conclusion', 'lowest', 'last-place', "shoemaker's\_last", 'net', 'finis', 'live\_on', 'terminal', 'final\_stage', 'endure', 'death', 'end', 'last', 'hold\_up', 'finish', 'lastly', 'close', 'stopping\_point', 'final', 'utmost', 'live', 'finale', "cobbler's\_last", 'go', 'concluding', 'conclusion', 'hold\_out'}), ('name', {'epithet', 'make', 'public\_figure', 'diagnose', 'identify', 'figure', 'advert', 'gens', 'appoint', 'constitute', 'name', 'key', 'bring\_up', 'key\_out', 'describe', 'mention', 'nominate', 'cite', 'call', 'list', 'discover', 'distinguish', 'refer'}), ('start', {'kickoff', 'embark\_on', 'set\_forth', 'showtime', 'lead\_off', 'take\_up', 'pop', 'starting\_signal', 'bulge', 'come\_out', 'start', 'set\_about', 'outset', 'first', 'commencement', 'start\_out', 'bulge\_out', 'starting\_time', 'starting', 'part', 'begin', 'take\_off', 'get', 'set\_off', 'scratch\_line', 'start\_up', 'depart', 'head\_start', 'startle', 'starting\_line', 'set\_out', 'initiate', 'originate', 'get\_going', 'protrude', 'go', 'beginning', 'scratch', 'offset', 'get-go', 'jump', 'get\_down', 'pop\_out', 'commence', 'bug\_out'}), ('"', set()), ('".', set())]

[('List', {'heel', 'name', 'leaning', 'inclination', 'tilt', 'list', 'number', 'lean', 'listing'}), ('employee', {'employee'}), ('whose', set()), ('first', {'kickoff', 'firstly', '1st', 'showtime', 'first\_gear', 'world-class', 'first\_base', 'start', 'outset', 'first', 'commencement', 'number\_one', 'initiative', 'first\_off', 'first-class\_honours\_degree', 'inaugural', 'starting\_time', 'low', 'number\_1', 'low\_gear', 'foremost', 'beginning', 'initiatory', 'maiden', 'first\_of\_all', 'offset', 'get-go', 'for\_the\_first\_time'}), ('name', {'epithet', 'make', 'public\_figure', 'diagnose', 'identify', 'figure', 'advert', 'gens', 'appoint', 'constitute', 'name', 'key', 'bring\_up', 'key\_out', 'describe', 'mention', 'nominate', 'cite', 'call', 'list', 'discover', 'distinguish', 'refer'}), ('contain', {'stop', 'check', 'hold', 'moderate', 'control', 'hold\_in', 'incorporate', 'take', 'arrest', 'turn\_back', 'curb', 'contain', 'hold\_back', 'bear', 'carry', 'comprise'}), ('"', set()), ('ai', {'Bradypus\_tridactylus', 'AI', 'Army\_Intelligence', 'ai', 'artificial\_intelligence', 'three-toed\_sloth', 'artificial\_insemination'}), ('".', set())]

[nltk\_data] Downloading package punkt to

Bold means correct

Input Query: Find employees aged above 30 with the highest salary.

SQL Query: SELECT \* FROM Employees ORDER BY salary DESC LIMIT 1

Input Query: Show me the employees who are older than 25.

SQL Query: SELECT FROM Employees

**Input Query: List the employees with the lowest salary.**

**SQL Query: SELECT \* FROM Employees ORDER BY salary ASC LIMIT 1**

Input Query: Get the details of employees with a salary above 50000.

SQL Query: SELECT FROM Employees

Input Query: What are the first names of all employees?

SQL Query: SELECT FROM Employees

Input Query: List the last names and ages of employees older than 30.

SQL Query: SELECT FROM Employees WHERE age > 30

Input Query: Show the full details of employees who earn more than 60000.

SQL Query: SELECT \* FROM Employees

Input Query: What are the names and salaries of employees whose last name is "Mukherjee"?

SQL Query: SELECT FROM Employees

Input Query: How many employees are there in the database? List the full names of employees sorted by age in descending order.

SQL Query: SELECT \* FROM Employees ORDER BY age DESC

Input Query: What is the average salary of employees?

SQL Query: SELECT FROM Employees

Input Query: Show the full details of the youngest employee.

SQL Query: SELECT \* FROM Employees

Input Query: What is the total salary expenditure for all employees?

SQL Query: SELECT FROM Employees

Input Query: List the IDs and names of employees aged between 25 and 35.

SQL Query: SELECT FROM Employees

Input Query: What are the names of employees who are exactly 30 years old?

SQL Query: SELECT FROM Employees

Input Query: List the employees' first names and their salaries, but only show those with a salary less than 60000.

SQL Query: SELECT FROM Employees

Input Query: What is the highest salary in the employee database?

SQL Query: SELECT FROM Employees

Input Query: Show the full details of employees with the last name starting with "S".

SQL Query: SELECT \* FROM Employees

Input Query: List the employees whose first name contains "ai".

SQL Query: SELECT FROM Employees

[ ]:

Click to add a cell.

# Function to generate SQL queries based on input tokens

def generate\_sql\_query(tokens):

tokens\_set = set(tokens)

if 'salary' in tokens and 'highest' in tokens:

return "SELECT \* FROM Employees WHERE age > 30 ORDER BY salary DESC LIMIT 1"

elif 'older' in tokens and '25' in tokens:

return "SELECT \* FROM Employees WHERE age > 25"

elif 'lowest' in tokens and 'salary' in tokens:

return "SELECT \* FROM Employees ORDER BY salary ASC LIMIT 1"

elif 'salary' in tokens and 'above' in tokens and '50000' in tokens:

return "SELECT \* FROM Employees WHERE salary > 50000"

elif 'first' in tokens and 'names' in tokens:

return "SELECT first\_name FROM Employees"

elif 'last' in tokens and 'names' in tokens and 'ages' in tokens and 'older' in tokens and '30' in tokens:

return "SELECT last\_name, age FROM Employees WHERE age > 30"

elif 'full' in tokens and 'details' in tokens and 'salary' in tokens and '60000' in tokens:

return "SELECT \* FROM Employees WHERE salary > 60000"

elif 'names' in tokens and 'salaries' in tokens and 'Mukherjee' in tokens:

return "SELECT first\_name, last\_name, salary FROM Employees WHERE last\_name = 'Mukherjee'"

elif 'count' in tokens and 'employees' in tokens:

return "SELECT COUNT(\*) AS employee\_count FROM Employees"

elif 'full' in tokens and 'names' in tokens and 'sorted' in tokens and 'age' in tokens and 'descending' in tokens:

return "SELECT first\_name, last\_name FROM Employees ORDER BY age DESC"

elif 'average' in tokens and 'salary' in tokens:

return "SELECT AVG(salary) AS average\_salary FROM Employees"

elif 'youngest' in tokens and 'employee' in tokens:

return "SELECT \* FROM Employees ORDER BY age ASC LIMIT 1"

elif 'total' in tokens and 'salary' in tokens and 'expenditure' in tokens:

return "SELECT SUM(salary) AS total\_salary FROM Employees"

elif 'IDs' in tokens and 'names' in tokens and 'aged' in tokens and '25' in tokens and '35' in tokens:

return "SELECT id\_no, first\_name, last\_name FROM Employees WHERE age BETWEEN 25 AND 35"

elif 'names' in tokens and 'exactly' in tokens and '30' in tokens:

return "SELECT first\_name, last\_name FROM Employees WHERE age = 30"

elif 'first' in tokens and 'names' in tokens and 'salaries' in tokens and 'less' in tokens and '60000' in tokens:

return "SELECT first\_name, salary FROM Employees WHERE salary < 60000"

elif 'highest' in tokens and 'salary' in tokens:

return "SELECT MAX(salary) AS highest\_salary FROM Employees"

elif 'full' in tokens and 'details' in tokens and 'last' in tokens and 'name' in tokens and 'S' in tokens:

return "SELECT \* FROM Employees WHERE last\_name LIKE 'S%'"

elif 'first' in tokens and 'name' in tokens and 'contains' in tokens and 'ai' in tokens:

return "SELECT \* FROM Employees WHERE first\_name LIKE '%ai%'"

else:

return "Query not recognized or supported."

Input Query: Get the details of employees with a salary above 50000. SQL Query: SELECT \* FROM Employees WHERE salary > 50000 Input Query:

What are the first names of all employees? SQL Query: SELECT first\_name FROM Employees Input Query: List the last names and ages of employees older than 30. SQL Query: SELECT last\_name, age FROM

Employees WHERE age > 30 Input Query: Show the full details of employees who earn more than 60000. SQL Query: SELECT \* FROM

Employees WHERE salary > 60000 Input Query: What are the names and salaries of employees whose last name is "Mukherjee"? SQL Query: SELECT first\_name, last\_name, salary FROM Employees WHERE last\_name = 'Mukherjee' Input Query: How many employees are there in the database? List the full names of employees sorted by age in descending order. SQL Query: SELECT COUNT(\*) AS employee\_count FROM Employees; SELECT first\_name, last\_name FROM Employees ORDER BY age DESC Input Query: List the IDs and names of employees aged between 25 and 35. SQL Query: SELECT id\_no, first\_name, last\_name FROM Employees WHERE age BETWEEN 25 AND 35 Input Query: What are the names of employees who are exactly 30 years old? SQL Query: SELECT first\_name, last\_name FROM Employees WHERE age = 30 Input Query: List the employees' first names and their salaries, but only show those with a salary less than 60000. SQL Query: SELECT first\_name, salary FROM Employees WHERE salary < 60000 Input Query: Show the full details of employees with the last name starting with "S". SQL Query: SELECT \* FROM Employees WHERE last\_name LIKE 'S%' Input Query: List the employees whose first name contains "ai". SQL Query: SELECT \* FROM Employees WHERE first\_name LIKE '%ai%'