/\* By putting DISTINCT after SELECT, you do not return duplicates.

***friends\_of\_pickles***

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **id** | **name** | **gender** | **species** | **height\_cm** |
| 1 | Dave | male | human | 180 |
| 2 | Mary | female | human | 160 |
| 3 | Fry | male | cat | 30 |
| 4 | Leela | female | cat | 25 |
| 5 | Odie | male | dog | 40 |
| 6 | Jumpy | male | dog | 35 |
| 7 | Sneakers | male | dog | 55 |

Consider the above table ‘friends\_of\_pickles’

\*/

-- 1. Can you return a list of the distinct species of animals greater than 50cm in height?

SELECT DISTINCT species FROM friends\_of\_pickles WHERE height\_cm > 50 ;

/\* If you want to sort the rows by some kind of attribute, you can use the ORDER BY keyword. It returns the attributes in ascending order.

In order to put the names in descending order, you would add a DESC at the end of the query. \*/

--2. Can you run a query that sorts the **friends\_of\_pickles** by height\_cm in descending order?

SELECT \* FROM friends\_of\_pickles ORDER BY height\_cm DESC ;

--LIMIT clause limits query results to a limited number of records

--3. Can you return the single row (and all columns) of the tallest **friends\_of\_pickles**?

SELECT \* from friends\_of\_pickles ORDER BY height\_cm DESC LIMIT 1 ;