2.)
Query 1: SELECT * FROM PERSONS;

  Cost:
    - 27
  Was it a full or index scan of the table?
    - Full scan
  Why?
    - Because there is no primary key constraint which oracle can use to automatically create
indexes.

3.)
Query 2: SELECT * FROM PERSONS WHERE person_id > 1000 AND person_id < 3000;

  Cost:
    - 27
  Was it a full or index scan of the table?
    - Full scan
  Why?
    - Yes. The result is outputted faster since there's less row of data being searched

4.)
Query 1: ALTER TABLE PERSONS
  ADD CONSTRAINT persons_pk PRIMARY KEY(person_id);
SELECT * FROM PERSONS;

  Cost:
    - 27
  Was it a full or index scan of the table?
    - Full scan
  Comment the results.
    - results is still the same as it is searching the whole table.

Query 2: SELECT * FROM PERSONS WHERE person_id > 1000 AND person_id < 3000;

  Cost:
    - 7
  Was it a full or index scan of the table?
    - Index scan
  Comment the results.
    - results are faster now since is uses index scan using the index that was automatically created
      when the primary key constraint was created.

5.)
Query 3: select * from persons where person_id+5>1000 and person_id<3000

  Cost:
   - 11
  Was it a full or index scan of the table?
   - Index scan
  Comment the results.
   - Query is still indexed

Query 4: select * from persons where person_id+5>1000 and person_id*2<3000

  Cost:
   - 27
  Was it a full or index scan of the table?
   - Full scan
  Comment the results.
   - Index is removed causing a full scan to be used

6.)
Query 5: select person_age, count(person_id) from persons group by person_age;

Cost:
   - 27
  Was it a full or index scan of the table?
   - Full scan
  Comment the results.
   - person_id does not help since we are grouping using person_age

7.)
Query 5 w/ index on person_Age:

Cost:
   - 9
  Was it a full or index scan of the table?
   - Index Scan

After modifying the table:

Cost:
   - 26
  Was it a full or index scan of the table?
   - Full Scan

8.)

Joining two tables:

Cost:
   - 239
   - Cardinality: 27
  Was it a full or index scan of the table?
   - Full Scan on Jobs and Full scan on Jobs_persons

After adding index on the primary/foreign keys:
   - Adding the index to the primary/foreign keys saw a reduction in cost due to using the indexes.