

```

INSERT INTO HIST_TABLE_GENERATION_MODELS A
SELECT TO_CHAR (a.date, 'DAY') AS Day_of_week,
       TO_CHAR (a.date, 'MM') AS Month,
       TO_CHAR (a.date, 'YYYY') AS Year,
       CASE
         WHEN TO_CHAR (a.date, 'DAY') = 'SUNDAY' THEN 'NOT WORKING'
         WHEN TO_CHAR (a.date, 'DAY') = 'SATURDAY' THEN 'NOT WORKING'
         ELSE 'WORKING'
       END
       AS Working,
       AVG (number_visits),
       AVG (Minimum_temperature),
       AVG (Mean_temperature),
       AVG (Maximum_temperature),
       AVG (Relative_humidity),
       SUM (Days_Bad_air_quality)
FROM data_of_visits a, environmental_data b, data_air_quality c
WHERE A.date = B.date AND a.date = C.date
GROUP BY TO_CHAR (a.date, 'DAY'),
         TO_CHAR (a.date, 'MM'),
         TO_CHAR (a.date, 'YYYY'),
         CASE
           WHEN TO_CHAR (a.date, 'DAY') = 'SUNDAY' THEN 'NOT WORKING'
           WHEN TO_CHAR (a.date, 'DAY') = 'SATURDAY' THEN 'NOT WORKING'
           ELSE 'WORKING'
         END;

```

Source SQL that makes the inclusion of grouped information

```

WITH "NAME_OF_PREDICTION"
AS (SELECT
      <ATTRIB>
    FROM
      <TABLE ORIGEN>
    WHERE
      ( CONDITIONS.. )),
"NAME_OF_MODEL" as (
  SELECT
    PREDICTION("REGR_GLM_1_1" USING *) "RESULT",
  FROM
    "NAME_OF_PREDICTION" )

```

Source SQL to call the predictive model

```

CREATE OR REPLACE TRIGGER TGR_MODELO_T_LAB
  BEFORE INSERT OR UPDATE
  ON RESULT
  FOR EACH ROW
BEGIN
  :NEW.FECHA_REALIZADO := SYSDATE;
  :NEW.DIA_SEMANA := LTRIM (RTRIM (TO_CHAR (:NEW.Date, 'DAY')));
  :NEW.MES := TO_CHAR (:NEW.Date, 'MM');
  :NEW.AÑO := TO_CHAR (:NEW.Date, 'YYYY');
  :NEW.MEDIA_CASOS := NULL;

  WITH "N$10086"
  AS (SELECT
        TO_CHAR (:NEW.DATE, 'MM') AS MONTH1,
        :NEW.AIR_QUALITY AS AIR_QUALITY ,
        SYSDATE AS DATE1,
        :NEW.Mean_temperature AS Mean_temperature,
        :NEW.Working AS Working,
        LTRIM(RTRIM(TO_CHAR(:NEW.DATE,'DAY'))) AS Day_of_week,
        :NEW.Relative_humidity AS Relative_humidity,
        TO_CHAR(:NEW.DATE,'YYYY') AS Year,
        :NEW.Maximum_temperature AS Maximum_temperature,
        :NEW.Maximum_temperature AS Maximum_temperature
      From DUAL),
  "N$10073" as (
  SELECT
    PREDICTION("TESIS"."REGR_GLM_2_1" USING *) "REGR_GLM_2_1_PRED" INTO
:NEW.PREDICTION
      FROM "N$10086" )
  select * from "N$10073";

END;
/

```

Source Trigger to call the predictive model

```

SELECT NULL, Year, number_visits
  FROM HIST_TABLE_GENERATION_MODELS
 WHERE   Year = (SELECT Year
                 FROM RESULT
                 WHERE ID = :P16_ID)
        AND Day_of_week = (SELECT Day_of_week
                            FROM RESULT
                            WHERE ID = :P16_ID_RECORD)

GROUP BY Year
ORDER BY Year;

```

Source code SQL that gets information to graph historical results

```
DBMS_DATA_MINING.CREATE_MODEL(  
  model_name           => "N$10073",  
  mining_function      => DBMS_DATA_MINING.REGRESSION,  
  data_table_name     => HIST_TABLE_GENERATION_MODELS,  
  case_id_column_name => '',  
  target_column_name  => 'number visits');
```

Source code SQL to regenerate the model again