

# Cas S2

1) Un taller que es dedica a realitzar revisions de vehicles conserva, per a cada revisió que realitza, una fitxa en paper amb la següent informació:

- NIF del propietari del vehicle
- Cognoms
- Nom
- Email
- Telèfon
- Matrícula del vehicle
- Marca
- Model
- Color
- Data de la revisió
- Kilometratge en el moment de la revisió
- Notes sobre la revisió

El propietari del taller decideix informatitzar la informació, per la qual cosa es requereix emmagatzemar l'esmentada informació en una base de dades relacional. Descriure les taules i columnes que s'utilitzaran per dissenyar un model de dades en 3a forma normal que permeti conservar la informació de manera adequada.

Informació addicional:

- Se suposa que el NIF d'una persona la identifica unívocament i no canvia mai.
- Se suposa que la matrícula d'un vehicle l'identifica unívocament, però pot canviar i és necessari per als mecànics disposar de la informació de totes les revisions que ha passat un determinat vehicle encara que hagi canviat la matrícula. Per tant, s'ha d'implementar un mètode senzill per aconseguir que els mecànics puguin identificar un vehicle encara que hagi canviat la seva matrícula.
- Un determinat vehicle pot canviar de propietari, però el que interessa és mantenir la informació del propietari actual per poder avisar-lo de la data de la pròxima revisió.
- Ja que el propietari del taller vol aplicar descomptes als clients habituals, és necessari que es pugui obtenir tota la informació de les revisions que ha encarregat un determinat client quan era el propietari d'un vehicle, encara que actualment el propietari del mateix vehicle sigui un altre client.

*Valor de la pregunta: 50% de la nota del cas*

2) En el sistema operatiu Linux s'utilitzen expressions regulars en utilitats com a grep, awk, vi, etc. per especificar patrons de text que cercar o substituir. Les implementacions dels motors d'expressions regulars no són totes iguals, per la qual cosa per al present exercici es prendrà un subconjunt de les expressions regulars esteses de GNU grep, definides de la següent manera:

- La majoria de caràcters, incloent els caràcters alfabètics i numèrics són expressions regulars que concorden amb ells mateixos. Es distingeixen les majúscules de les minúscules.
- El punt (".") concorda amb qualsevol caràcter individual.
- Les expressions de claudàtors concorden amb qualsevol caràcter individual que continguin. Per exemple, "[afj]" concorda amb la lletra "a", la "f" o la "j". Es poden

especificar rangs amb "-" (per exemple, "[a-zA-Z]"). Es pot fer que l'expressió concordi amb els caràcters que NO conté utilitzant el símbol "^" com a primer caràcter (per exemple, "[^a-z]"). Per incloure el caràcter "^" sense que s'apliqui el seu significat especial, s'ha de situar en qualsevol lloc entre els claudàtors excepte la primera posició ("^[a-z]"). Per incloure el caràcter "-" sense que s'apliqui el seu significat especial, s'ha de situar entre els claudàtors en la darrera posició ("[a-z-]"). Per incloure el caràcter "]" sense que es tanqui l'expressió de claudàtors, s'ha de situar just després del claudàtor d'obertura inicial ("[]a-z").]

- Els símbols "^" i "\$" són metacaràcters que concorden amb una cadena buida a l'inici i final de la línia respectivament.
- El símbol "?" fa que l'element anterior sigui opcional, és a dir, fa que concordi si apareix 0 o 1 vegades.
- El símbol "\*" concorda quan l'element anterior està repetit 0 o més vegades.
- El símbol "+" concorda quan l'element anterior està repetit 1 o més vegades.
- Dues expressions regulars concatenades concorden amb qualsevol cadena formada per dues subcadena concatenades que concorden respectivament amb cada una de les dues expressions regulars.
- L'operador "|" es pot utilitzar per especificar alternança. Per exemple "uno|dos|tres" concordaria amb qualsevol de les tres paraules.
- La precedència dels operadors s'aplica de la següent manera: en primer lloc, els operadors de repetició ("?", "\*", "+"); en segon lloc, la concatenació; i en tercer lloc l'alternança ("|"). La precedència es pot modificar utilitzant els parèntesis. Així l'expressió regular "(ab\*|cd\*|ef\*)+", concordaria amb qualsevol cadena composta d'un o més elements formats de la següent manera: una "a" seguida d'un nombre indeterminat de "b" (incloent 0) o bé una "c" seguida d'un nombre indeterminat de "d" (incloent 0) o bé una "e" seguida d'un nombre indeterminat de "f" (incloent 0). Un exemple podria ser "abbbbacddddcaefffffffffff".
- Per fer concordar els caràcters especials (".", "\*", etc.) amb el caràcter literal i no aplicar l'operació especial que implica, s'utilitza la barra inversa (també per a la pròpia barra inversa). Per exemple, "google\\.com" o "c:\\windows".
- Per simplicitat, només es considera un joc de caràcters ASCII de 7 bits, de tal manera que, per exemple, "[a-z]" concorda amb totes les lletres minúscules, ja que no es tenen en compte altres "lletres" com caràcters accentuats, específics d'idiomes concrets ("ñ", "ç"), etc., que requereixen tractament especial per part del motor d'expressions regulars que s'utilitza.

Així doncs, es requereix crear expressions regulars, segons el subconjunt de funcionalitat definit, per als casos que es plantegen:

- a) Crear una expressió regular que concordi amb les diferents formes de la paraula "gato": en masculí o femení i les seves respectives versions en plural. (Valor: 7,14%)
- b) Crear una expressió regular que concordi amb un nom de servidor en el domini "prueba.com". Es considera que els noms de servidor abans del domini poden contenir caràcters alfanumèrics en minúscules i el guió, però no poden començar per guió (per exemple, "servidor-web-1.prueba.com"). (Valor: 7,14%)

- c) Crear una expressió regular que concordi amb el format general d'una adreça IP amb els components individuals de l'adreça sempre compostos per 3 xifres entre 0 i 9 (no té per què ser una adreça IP vàlida, ja que pot contenir components majors que 255). Per exemple: "010.850.000.305". (Valor: 7,14%)
- d) Crear una expressió regular que concordi amb les lletres "abcd" al principi d'una línia. Qualsevol lletra haurà de poder aparèixer en majúscula o minúscula indistintament. Per exemple "AbcD". (Valor: 7,14%)
- e) Crear una expressió regular que concordi amb un camí absolut de Windows (per exemple, "c:\Windows\system32". Es poden utilitzar majúscules, minúscules i dígit. La lletra d'unitat inicial seguida dels dos punts és opcional. Cada component del camí ha d'estar compost per caràcters alfanumèrics (no s'accepta, per exemple, "." o "..") i ha de tenir almenys un caràcter (per tant, no poden aparèixer dues barres inverses seguides). El camí no pot acabar en una barra inversa, excepte si es tracta del directori arrel de la unitat. (Valor: 7,14%)
- f) Crear una expressió regular que concordi amb una data i hora en format "DD/MM/AAAA HH:mm:ss". Cada component ha de tenir el nombre de xifres especificat, omplint amb zeros a l'esquerra si és necessari. L'expressió ha de permetre dies només entre "00" i "31" (independentment del nombre de dies que tingui el mes), mesos només entre "01" i "12", anys a partir del 2000, hores entre "00" i "23", minuts entre "00" i "59" i segons entre "00" i "60" (els segons poden arribar fins a 60 en lloc de fins a 59 en ocasions especials). (Valor: 7,14%)
- g) Crear una expressió regular que concordi amb una línia composta únicament i exclusivament per dígit senars (com a mínim 3 dígit). (Valor: 7,14%)

*Valor de la pregunta: 50% de la nota del cas*