



ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΕΙΡΑΙΩΣ
UNIVERSITY OF PIRAEUS

Μάθημα: «Βάσεις Δεδομένων (4ο εξ.)»

Ομάδα εργασίας:
ΑΜ:Ρ14086, Κουσουννής Κωνσταντίνος
ΑΜ:Ρ14036, Γιώργος Διώτης

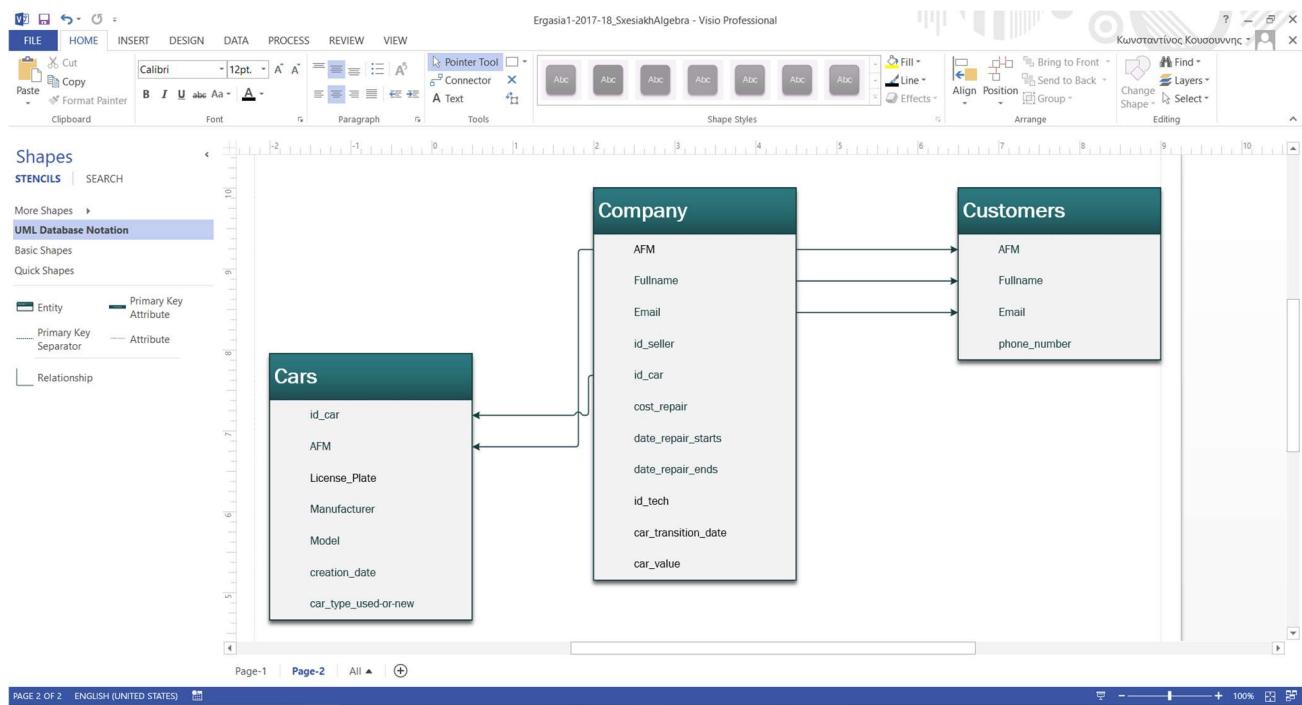
Ερώτημα 1 (30%). Σχεσιακή Βάση Δεδομένων

(a) Με βάση τα παραπάνω στοιχεία, σχεδιάστε το σχεσιακό σχήμα της ΒΔ και υλοποιήστε το (εντολές CREATE TABLE) στο ΣΔΒΔ PostgreSQL. Επιπλέον, καλείστε να τεκμηριώσετε τους

περιορισμούς ακεραιότητας των πινάκων, καθώς και να φορτώσετε με δεδομένα τους πίνακες. Το παραδοτέο του υποερωτήματος είναι το σχεσιακό σχήμα της ΒΔ, οι εντολές CREATE TABLE και τα αρχεία τα οποία θα εισάγετε στους πίνακες. Οδηγία: για την ευκολότερη παραγωγή αληθοφανών δεδομένων προτείνεται να χρησιμοποιήσετε κάποιο data generator (π.χ. www.mockaroo.com).

(b) Εφαρμόστε τη θεωρία της κανονικοποίησης πάνω στο σχεσιακό σχήμα της ΒΔ που σχεδιάσατε και ελέγξτε εάν ακολουθεί την 3NF ή BCNF.

Αρχικά ξεκινάω με βάση την εκφώνηση να σχεδιάζω την βάση δεδομένων μου με το προγράμμα Visio για να σχεδιάσω.



Αρχικά ξεκινάω με την Company (Εταιρία).

Company	
AFM	
Fullname	
Email	
id_seller	
id_car	
cost_repair	
date_repair_starts	
date_repair_ends	
id_tech	
car_transition_date	
car_value	

Εδώ βάζω τα υποχρετικά πεδία που μου λέει αυτά είναι αρχίκα για την εταιρεία ΑΦΜ(αριθμό μητρώου ασφαλιζομένου) ολοκληρω το όνομα(Fullname) το email>Email).

Μετά για τους τεχνικούς ζητάει το id του κάθε τεχνικού(id_tech) το id του αυτικινήτου(id_car) το κόστος της επισκευής (cost_repair) την ημερομηνία που ξεκινάει(date_repair_starts) η επισκευή και την ημερομηνία που τελειώνει η επισκευεί(date_repair_ends).

Τέλος για τους πωλητές ζητάει το id του κάθε πωλήτη(id_seller) τα στοιχεία του αυτοκινήτου(id_car) που αγοραστήκε ή πωλήθηκε, την αξία αγοράς ή πώλησης(car_value), ημερομηνία αγοράς ή πώλησης(car_transition_date)

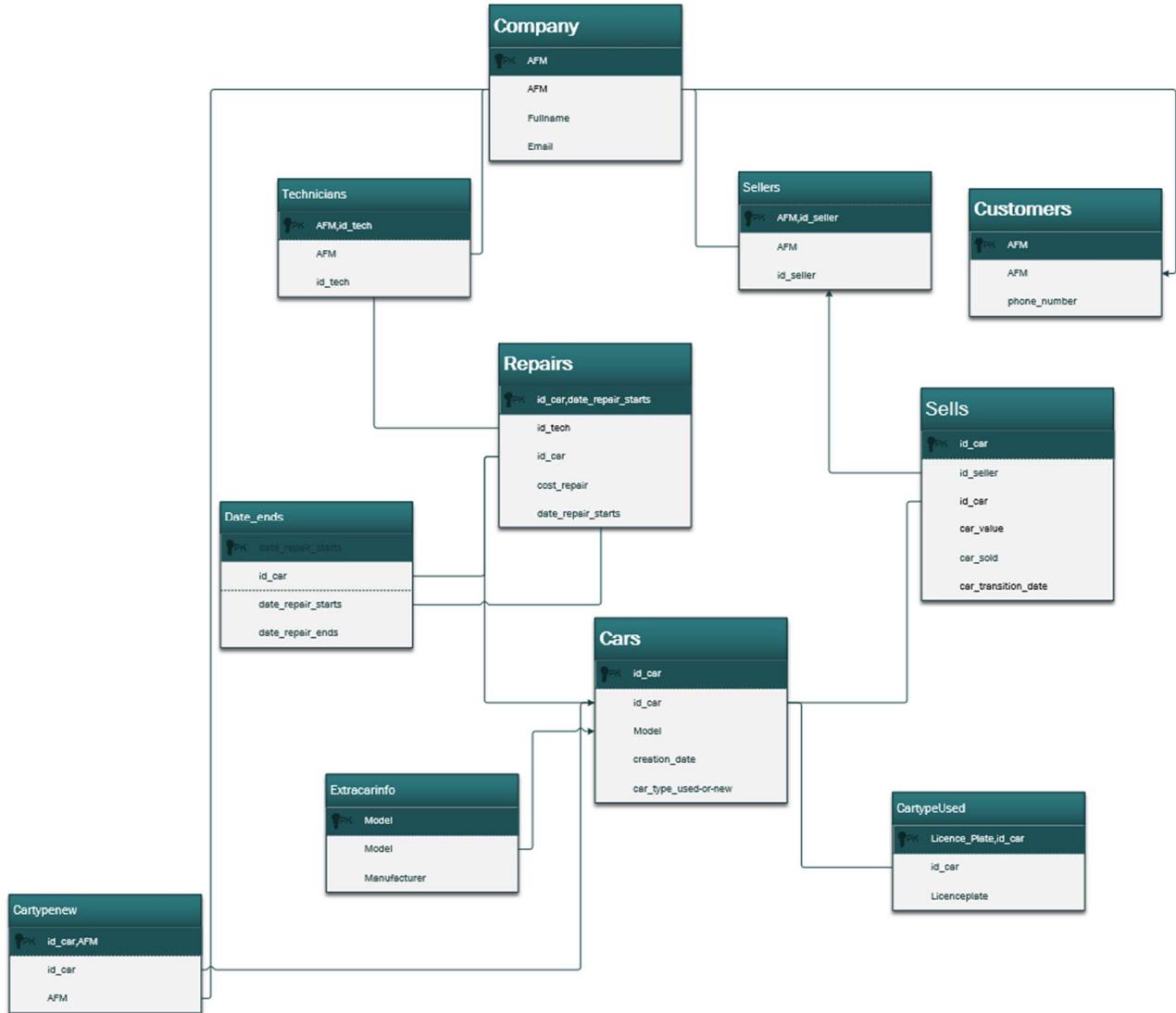
Cars
id_car
AFM
License_Plate
Manufacturer
Model
creation_date
car_type_used-or-new

Στην συνέχεια φτιάχνουμε τον πίνακα cars ο οποίος πρέπει υποχρεωτικά να έχει τον αριθμό κυκλοφορίας(Licence_Plate) τα στοιχεία του ιδιοκτήτη(AFM) κατασκευαστής (manufacturer) το μοντέλο (model) την ημερομηνία παραγωγής (creation_date) και το είδος του αυτοκινήτου μεταχειρισμένο καινουργιό (car_type_used-or-new)

Τέλος κατασκευάζουμε τον πίνακα πελάτες ο οποίος περιέχει τον αριθμό μητρώου (AFM) το ολοκληρωτικό όνομα (fullname)

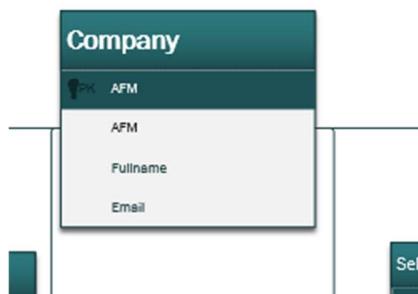
Customers
AFM
Fullscreen
Email
phone_number

Στην συνέχεια όπως παρατηρώ η βάση δεδομένων μου δεν είναι σωστή αυτην την στιγμή θα πρέπει να σπάσω τον πίνακα company έτσι ώστε να επιτευχθεί η ακεραιότητα των πινάκων διοτι αυτην την στιγμή ο πινακας company δεν είναι σωστός υπάρχουν στηλες που δεν χρησιμοποιούνται εχουμε πλεονασμο πληροφοριας.



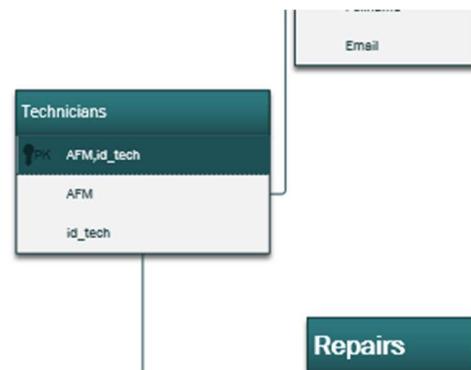
Παραπάνω εχουμε το νεο σχεσιακο σχημα το οποιο τηρει ολες τις προυποθεσεις

Αναλυτικα:



Εχουμε τον πινακα **Company** ο οποιος περιεχει το **AFM** όλων(το οποιο ειναι και το **πρωτευον κλειδι**) που σχετιζονται με την εταιρια δηλαδη τους τεχνικους του πωλητές και τους πελατες τις,το ολοκληρωμενο ονομα **Fullname** όλων και τα **email** τους.

Μετα εχουμε εναν πινακα που λεγεται **Technicians**
ο οποις ειναι ενας πινακας ο οποιος περιεχει το **AFM** του
καθε τεχνικού και το id του **id_tech** εχει **πρωτευον κλειδί**
τον συνδιασμο **AFM,id_tech**



Στην συνέχεια εχουμε τον πινακα **Sellers** ο οποιος περιέχει το **AFM**
και το id του καθε πωλητη **id_seller** εχει **πρωτευον κλειδί** τον
συνδιασμο **AFM,id_seller**



Εχουμε τον πινακα **Customers** πελατες ο οποιος περιεχει το **AFM**
των πελατων και τον αριθμο τηλεφωνου **phone_number** και
πρωτευον κλειδι ειναι το **AFM**.



Εχουμε τον πινακα **Repairs** ο οποίος περιεχει το **id_tech** το id των
τεχνικών που επισκευαζουν τα αυτοκινητα θα εχουμε επαναληψη
τεχνικων σε αυτον τον πινακα,**id_car** το id του αυτοκινητου που μπηκε
για επισκευη και εδω μπορει να ξανα ερθει ενα αυτοκινητο,
cost_repair το κοστος επιδιορθωδης ,**date_repair_starts** η ημερομηνια
που ξεκινάει η επισκευή.Το **πρωτευον κλειδι** σε αυτο τον πινακα ειναι
ο συνδιασμός του **id_car,date_repair_starts**



Εχουμε τον πίνακα **Sells** ο οποιος εχει τις πωλησεις των αυτοκινητων περιέχει το **id_seller** το id του πωλητη και το **id_car** το id του αυτοκινητου ,το **car_value** αξια αυτοκινητου ,**car_sold** πωληση αυτοκινητου και **car_transition_date** την ημερα που μεταβιβαστηκε το αυτοκινητο.Το **πρωτευον κλειδι** ειναι το **id_car**



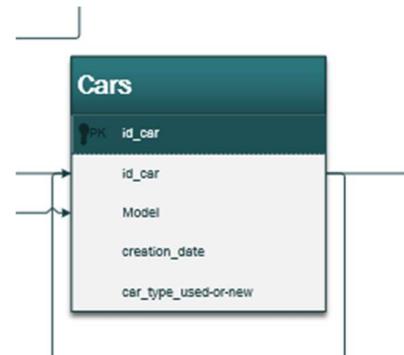
Date_ends

PK	date_repair_starts
	id_car
	date_repair_starts
	date_repair_ends

Στην συνεχεια εχω εναν πίνακα **Date_ends** ο οποιος περιέχει την ημερομηνία που τελειώνει η επισκευη του αυτοκινητου στην ουσια εσπασα τον πίνακα repairs για να μην εχω κενα στοιχεια στον πίνακα γιατι εχω ημερομηνιες που δεν εχει συμπλειρωθει ακομα η ημερα που τελειωσε η επισκευη αρα σε αυτο τον πίνακα εχω μονο τις ημερομηνιες που εχουν ημερομηνια που τελειωνει η επισκευή.Αυτος ο πίνακας περιέχει **id_car** το id του αυτοκινητου,**date_repair_starts** την ημερομηνια που τελειωσε η επισκευη **date_repair_ends**,το **πρωτευον κλειδι** ειναι ο **συνδιασμός id_car,date_repair_starts**.

Πινακας **Cars** εχω το **id_car** το id του αυτοκινήτου ,**model** το μοντέλο του αυτοκινήτου,**creation_date** ημερα

δημιουργιας ,**car_type_used_or_new** αν το αυτοκινητο ειναι μεταχειρισμενο ,καινουργιο και αν δεν εχει πουληθει ακομα.Το **πρωτευον κλειδι** ειναι το **id_car**

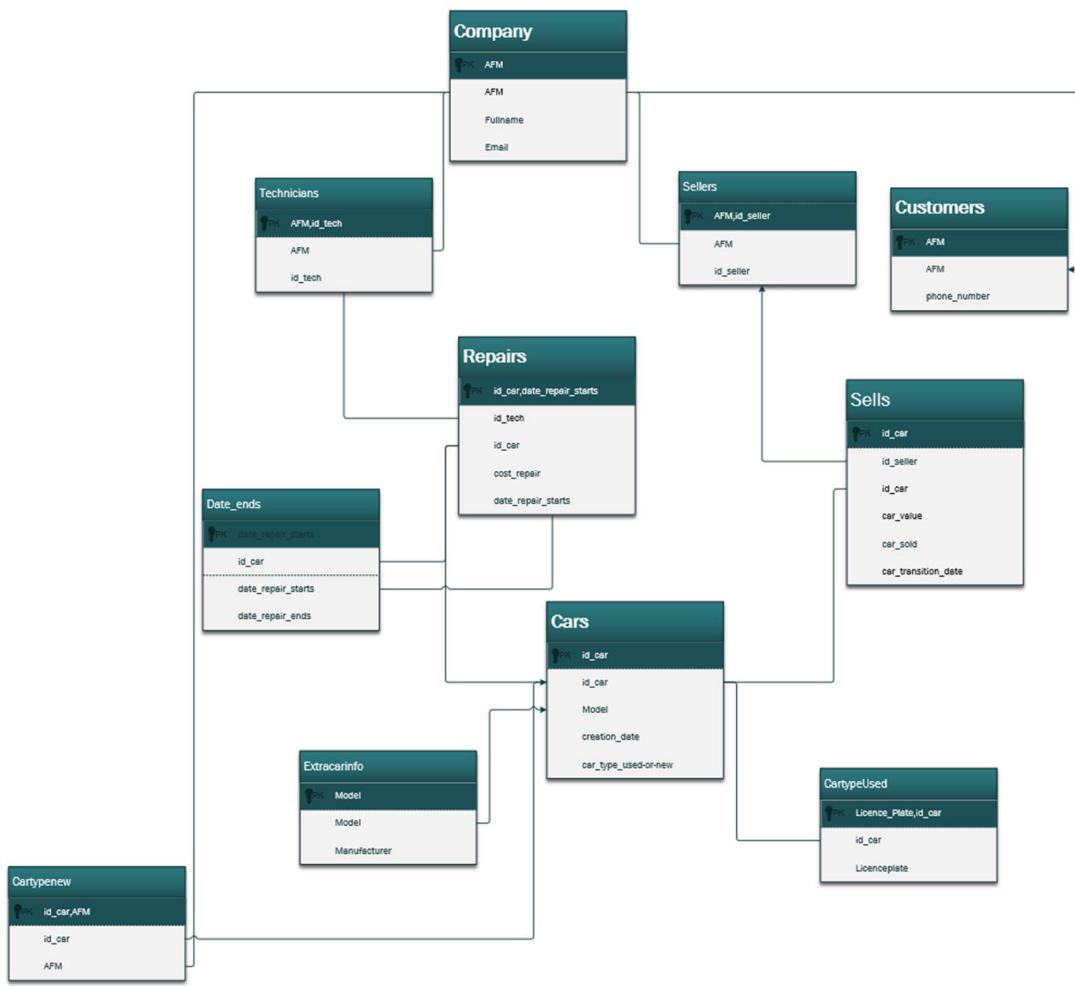


Εχουμε τον πινακα **Extracarinfo** ο οποιος στην ουσια ηταν τμημα του πινακα cars αλλα εσπασα τον πινακα γιατι το μοντέλο και ο κατασκευαστης συνδεόνται μεταξύ τους.Οποτε έχουμε **Model** το μοντέλο και τον **Manufacturer** και τον κατασκευαστή.

Πινακας **CartypeUsed** που δινει πληροφορίες αναλογα με τον τυπο του αυτοκινήτου δηλαδή αν απο τον πίνακα cars το χαρακτηριστικό car_type_used-or-new ειναι used τοτε θα εχει πινακίδα κυκλοφορίας **id_car** το id αυτοκινήτου ,**Licenceplate** ειναι η πινακίδα κυκλοφορίας με πρωτεύον κλειδι την πινακίδα κυκλοφορίας.



Πινακας **Cartypenew** και αυτος ο πινακας περιεχει πληροφοριες αναλογα με το ειδος του αυτοκινητου απο τον πινακα cars αυτος ο πινακας δηλαδη περιέχει τα καινουργια αυτοκινητα που εχουν ιδιοκτητη.**id_car** το id του αυτοκινητου και **AFM** το afm των ιδιοκτητων. **Πρωτευον κλειδι** ειναι ο συνδιασμός το **id_car** και το **AFM**.



Οπως βλεπουμε εχει γινει κανονικοποιηση των προηγουμενων πινακων και αυτο συμβαίνει γιατι:

- 1)Δεν εχουμε απώλεια πληροφορίας
- 2)Δεν εχουμε πλεονασμό πληροφοριας η πινακες εχουν σπασει ετσι ακριβως ωστε να μην εχουμε επαναληψη σε δεδομενα αλλα ολα τα δεδομενα ειναι ξεχωριστα μεταξυ τους.
- 3)Διατήρηση εξαρτήσεων ολοι οι πινακες συνδεονται μεταξύ τους.

Το σχεσιακο σχήμα της βασης δεδομένων μου ειναι.

Δημιουργία δεδομενων με την βοηθεια του mockaroo
<https://mockaroo.com/>

Mockaroo - Random Data

Secure | https://mockaroo.com

mockarоо realistic data generator

PRICING SIGN IN

Need some mock data to test your app? Mockaroo lets you generate up to 1,000 rows of realistic test data in CSV, JSON, SQL, and Excel formats. Download data using your browser or sign in and create your own Mock APIs. Need more data? Plans start at just \$50/year.

Field Name	Type	Options
id	Row Number	blank: 0 % <input type="button" value="fx"/> <input type="button" value="x"/>
first_name	First Name	blank: 0 % <input type="button" value="fx"/> <input type="button" value="x"/>
last_name	Last Name	blank: 0 % <input type="button" value="fx"/> <input type="button" value="x"/>
email	Email Address	blank: 0 % <input type="button" value="fx"/> <input type="button" value="x"/>
gender	Gender	blank: 0 % <input type="button" value="fx"/> <input type="button" value="x"/>
ip_address	IP Address v4	blank: 0 % <input type="button" value="fx"/> <input type="button" value="x"/>

Add another field

Rows: 1000 Format: CSV Line Ending: Unix (LF) Include: header BOM

More ▾ Want to save this for later? [Sign up for free.](#)

https://mockaroo.com

Εφτιαξα τους αντιστοιχους πινακες και τους εκανα download

	Name	Date modified	Type	Size
	Cars	5/24/2018 12:16 PM	Microsoft Excel W...	42 KB
	Cartypenew	5/23/2018 5:01 PM	Microsoft Excel W...	27 KB
	CartypeUsed	5/23/2018 3:18 PM	Microsoft Excel W...	11 KB
	Company	5/28/2018 1:11 PM	Microsoft Excel W...	56 KB
	Customers	5/23/2018 3:24 PM	Microsoft Excel W...	32 KB
	Date_ends	5/27/2018 3:39 AM	Microsoft Excel W...	57 KB
	Extracarinfo	5/23/2018 3:20 PM	Microsoft Excel W...	21 KB
	Repairs - Copy	5/27/2018 3:31 AM	Microsoft Excel W...	74 KB
	Repairs	5/27/2018 11:39 A...	Microsoft Excel W...	97 KB
	Sellers	5/23/2018 3:28 PM	Microsoft Excel W...	11 KB
	Sells	5/24/2018 4:01 PM	Microsoft Excel W...	59 KB
	Technicians	5/23/2018 1:46 PM	Microsoft Excel W...	12 KB

Τα εχω αποθηκευση δυο φορες μια σε μορφη book και μια σε μορφη csv η μορφη csv ειναι για να περασω τα δεδομενα μου στην postgres

Σημαντικό οπου εχουμε ημερομηνιες εχω κανει πανω στην στηλη format για να αλλαξω την σειρα της ημερομηνιας yyyy-mm-dd

Εχουμε τον πινακα company

A	B
1 AFM	Fullname
2 53315050 Cyrille Garm	cgarms0@blogs.com
3 12530189 Emelen Lamprecht	elamprecht1@spfnn.com
4 51748644 Karol Crumme	kcrumme2@stanford.edu
5 97330761 Petronilla Robillart	probillart3@animotion.com
6 93276961 Rickert Ebbens	rebbens4@google.com.hk
7 76929065 Peria Goldby	pgoldby5@clickbank.net
8 90050585 Eydie O' Donohoe	eod6@diigo.com
9 66571260 Dewey Mcasparan	dmcasparan7@howstuffworks.com
10 94891468 Shellysheldon Wakefield	swakefield8@about.com
11 45300091 Elwin Isack	eisack9@webs.com
12 95851113 Ty Carletto	tcarletto10@mail.ru
13 67414367 Malena Bramo	mbramolo11@netvibes.com
14 52504726 Tallou Neljes	tneljesc12@devhub.com
15 32319654 Cristal Keave	ckeaveled13@toplist.cz
16 74656951 Herman Scane	hscaneee@evernivation.com
17 92018458 Prudy McMurdo	pmcmurdo1@livejournal.com
18 68680761 Kellie Braybrooks	kbraybrooks1@cbc.ca
19 26232015 Lorin Archbell	larchbell1@omniture.com
20 70220328 Emylee Waine	ewainei1@shareasale.com
21 28083395 Easter Isabell	eisabellj1@buzzfeed.com
22 66168165 Drusy Proven	dprovenk12@xrea.com
23 27681831 Davie Von Brook	dvonl1@facebook.com
24 48817323 Nanci Baguley	nbaguleym1@php.net
25 80931880 Irma Lamborre	ilambornen1@opensource.org
26 83045092 Gino Boomes	gboomeso1@lycos.com
27 96402874 Baudoin Casaccio	bcasacciol1@vistaprint.com
28 10758422 Garrard Ledamun	gledamunq1@europa.eu
29 31265553 Shaw Detoc	sdetocr1@kontakte.ru

Technicians

Sellers

A	B
74	32557858 S072
75	7341589 S073
76	78941362 S074
77	41458649 S075
78	54510195 S076
79	96723562 S077
80	17412063 S078
81	46466828 S079
82	79783169 S080
83	50211153 S081
84	34566889 S082
85	63442632 S083
86	78857300 S084
87	98385922 S085
88	33814166 S086
89	29703226 S087
90	5116969 S088
91	4805184 S089
92	40925015 S090
93	95168474 S091
94	77200899 S092
95	62444774 S093
96	92862347 S094
97	60474798 S095
98	59775315 S096
99	2566375 S097
100	42706346 S098
101	9161375 S099
102	

Customers

Repairs

A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T
1	T014	C0606	1419	2016-04-23															
2	T090	C0137	1500	2017-05-21															
3	T007	C0833	1585	2012-04-13															
4	T056	C0286	821	2015-07-05															
5	T030	C0465	856	2012-07-17															
6	T056	C0483	1360	2014-07-01															
7	T067	C0316	1419	2016-06-14															
8	T092	C0058	803	2010-07-28															
9	T099	C0118	1523	2016-10-24															
10	T012	C0442	1248	2015-01-15															
11	T035	C0018	1554	2013-08-14															
12	T016	C0610	472	2013-09-05															
13	T083	C0137	757	2016-04-30															
14	T032	C0660	1792	2016-11-23															
15	T046	C0904	398	2015-08-04															
16	T029	C0270	437	2011-08-20															
17	T075	C0152	814	2012-05-11															
18	T022	C0101	323	2015-08-27															
19	T029	C0985	316	2015-06-14															
20	T086	C0233	206	2015-12-25															
21	T057	C0066	1955	2011-01-07															
22	T096	C0326	628	2015-01-30															
23	T088	C0517	285	2014-02-22															
24	T057	C0897	1861	2011-01-18															
25	T052	C0092	1812	2014-10-25															
26	T030	C0366	288	2014-10-26															
27	T080	C0100	1599	2014-10-27															
28	T002	C0305	335	2014-10-28															

Sells

jobid	id_seller	id_car	car_value	car_sold	car_transition_date
1	S097	C0000	14976	15576	2010-11-01
2	S054	C0001	11133	11733	2010-03-12
3	S078	C0002	12803	13403	2010-06-11
4	S029	C0003	16014	16614	2010-10-13
5	S047	C0004	17274	17874	2010-04-27
6	S099	C0005	16225	16825	2010-03-24
7	S020	C0006	10845	11445	2010-10-08
8	S093	C0007	11375	11975	2011-01-07
9	S045	C0008	10960	11160	2010-07-02
10	S023	C0009	18616	18816	2011-01-21
11	S033	C0010	19999	20199	2010-10-22
12	S039	C0011	11185	11385	2010-12-13
13	S015	C0012	11407	11607	2010-08-28
14	S061	C0013	17571	17771	2010-08-17
15	S051	C0014	15058	15258	2010-12-11
16	S079	C0015	13868	14068	2010-07-17
17	S036	C0016	14518	14718	2010-07-21
18	S082	C0017	13380	13580	2010-05-29
19	S010	C0018	16309	16509	2010-05-22
20	S089	C0019	15107	15307	2010-07-08
21	S042	C0020	18071	18271	2010-05-24
22	S063	C0021	18005	18205	2010-10-16
23	S006	C0022	14412	14612	2010-12-12
24	S074	C0023	17700	17900	2010-04-02
25	S056	C0024	16686	16886	2010-07-13
26	S071	C0025	17130	18130	2010-10-18
27	S049	C0026	13667	14667	2010-06-12
28	S062	C0027	14812	15812	2010-03-28

Date_ends

Date_ends - Excel

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U
1	id_car	date_repair_starts	date_repair_ends																		
2	C0606	2016-04-23	2016-04-29																		
3	C0137	2017-05-21	2017-05-27																		
4	C0833	2012-04-13	2012-04-19																		
5	C0286	2015-07-05	2015-07-11																		
6	C0465	2012-07-17	2012-07-23																		
7	C0483	2014-07-01	2014-07-07																		
8	C0316	2016-06-14	2016-06-20																		
9	C0058	2010-07-28	2010-08-03																		
10	C0118	2016-10-24	2016-10-30																		
11	C0442	2015-01-15	2015-01-21																		
12	C0018	2013-08-14	2013-08-20																		
13	C0610	2013-09-05	2013-09-11																		
14	C0137	2016-04-30	2016-05-06																		
15	C0660	2016-11-23	2016-11-29																		
16	C0904	2015-08-04	2015-08-10																		
17	C0270	2011-08-20	2011-08-26																		
18	C0152	2012-05-11	2012-05-17																		
19	C0101	2015-08-27	2015-09-02																		
20	C0985	2015-06-14	2015-06-20																		
21	C0233	2015-12-25	2015-12-31																		
22	C0066	2011-01-07	2011-01-13																		
23	C0326	2015-01-30	2015-02-05																		
24	C0517	2014-02-22	2014-02-28																		
25	C0897	2011-01-18	2011-01-24																		
26	C0092	2014-10-25	2014-10-25																		
27	C0366	2014-10-26	2014-10-26																		
28	C0100	2014-10-27	2014-10-27																		
29	C0305	2014-10-28	2014-10-28																		

Cars

Cars - Excel

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U
1	id_car	Model	creation_date	car_type_used_or_new																	
2	C0000	Cirrus	2009-09-27	newscelled																	
3	C0001	Mulsanne	2009-02-05	newscelled																	
4	C0002	4000s Qua	2009-05-07	newscelled																	
5	C0003	Sonata	2009-09-08	used																	
6	C0004	M6	2009-03-23	newscelled																	
7	C0005	Tempest	2009-02-17	newscelled																	
8	C0006	tC	2009-09-03	newscelled																	
9	C0007	Expo	2009-12-03	newscelled																	
10	C0008	Tahoe	2009-05-28	newscelled																	
11	C0009	Durango	2009-12-17	used																	
12	C0010	Bonneville	2009-09-17	newscelled																	
13	C0011	Montana	2009-11-08	newscelled																	
14	C0012	Cobalt	2009-07-24	newscelled																	
15	C0013	Camaro	2009-07-13	used																	
16	C0014	Seville	2009-11-06	newscelled																	
17	C0015	XJ Series	2009-06-12	newscelled																	
18	C0016	GL-Class	2009-06-16	newscelled																	
19	C0017	Enclave	2009-04-24	newscelled																	
20	C0018	Hombre Sg	2009-04-17	used																	
21	C0019	HHR	2009-06-03	newscelled																	
22	C0020	Ram 1500	2009-04-19	newscelled																	
23	C0021	Caliber	2009-09-11	used																	
24	C0022	98	2009-11-07	newscelled																	
25	C0023	Familia	2009-02-26	newscelled																	
26	C0024	E-Class	2009-06-08	newscelled																	
27	C0025	Stylus	2009-09-13	newscelled																	
28	C0026	Express 35	2009-05-08	newscelled																	
29	C0027	Pajero	2009-02-21	used																	

Extracarinfo

Extracarinfo - Excel

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W
1	Model	Manufacturer																					
2	100	Quamba																					
3	1000	Omiba																					
4	1500	Trilia																					
5	1500 Club	Mydo																					
6	164	Thoughtstorm																					
7	200	Yoveo																					
8	228	Skinix																					
9	240	Centimia																					
10	2500	Talane																					
11	3 Series	Ooba																					
12	3-Sep	Realcube																					
13	300D	Yadel																					
14	300M	Browselab																					
15	300SE	Yakitri																					
16	300ZX	Twimm																					
17	323	Edgepulse																					
18	325	Fiveclub																					
19	3500	Babblestorm																					
20	3500 Club	Vimbo																					
21	350Z	Thoughtworks																					
22	4000s	Blogtag																					
23	4000s Qua	Podcat																					
24	400E	InnoZ																					
25	4Runner	Myworks																					
26	5 Series	Rhynoodle																					
27	5-Sep	Thoughtblab																					
28	57	Flashset																					
29	6 Series	Brightdog																					

CartypeUsed

CartypeUsed - Excel

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W
1	A d_car	Licenceplate																					
2	C0003	OVX5733																					
3	C0009	PSB2045																					
4	C0013	IYM7064																					
5	C0018	YLD1983																					
6	C0021	BWF1374																					
7	C0027	EIR5315																					
8	C0037	UFN9498																					
9	C0039	HZ03708																					
10	C0041	CAI4363																					
11	C0051	LVE8052																					
12	C0058	NVX4755																					
13	C0061	EMU1240																					
14	C0071	BLX3199																					
15	C0081	XVE1253																					
16	C0100	QOL6794																					
17	C0103	MI06555																					
18	C0115	GAP6608																					
19	C0135	PIW5910																					
20	C0170	DOL7792																					
21	C0177	EPX9587																					
22	C0179	ATC1585																					
23	C0182	TSF9747																					
24	C0193	WMB2826																					
25	C0196	LI04302																					
26	C0200	OWX2929																					
27	C0209	WNE7138																					
28	C0213	TLX2266																					
29	C0217	JS07034																					

Cartypenew

Cartypenew - Excel

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W
1	A d_car	AFM																					
2	C0000	53315050																					
3	C0001	12530189																					
4	C0002	51748644																					
5	C0004	93276961																					
6	C0005	76929065																					
7	C0006	90050585																					
8	C0007	66571260																					
9	C0008	94891468																					
10	C0010	95851113																					
11	C0011	67414367																					
12	C0012	52504726																					
13	C0014	74656951																					
14	C0015	92018458																					
15	C0016	68680761																					
16	C0017	26232015																					
17	C0019	28083395																					
18	C0020	66186185																					
19	C0022	48817323																					
20	C0023	80931880																					
21	C0024	83045092																					
22	C0025	96402874																					
23	C0026	10758422																					
24	C0028	56534685																					
25	C0029	83665305																					
26	C0030	106303																					
27	C0031	43883489																					
28	C0032	94497405																					
29	C0033	96334403																					

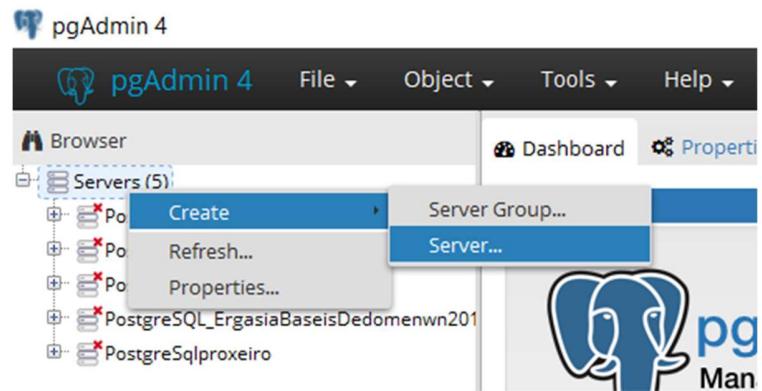
Αρχικα θα πω από και θα ανοιξω την Postgresql και θα φτιάξω μια βαση δεδομένων



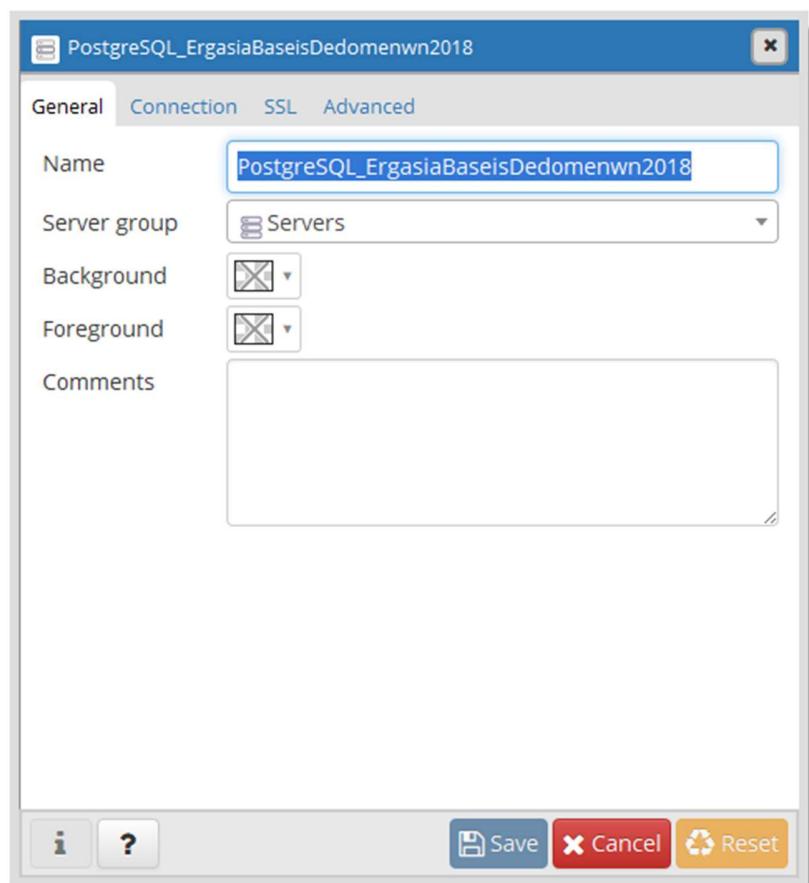
To ανοιγω με διπλο αριστερο click

Και θα πω να φτιαξω την βαση δεδομενων μου

Θα πω πανω απο τον server θα πατησω δεξι click create→server

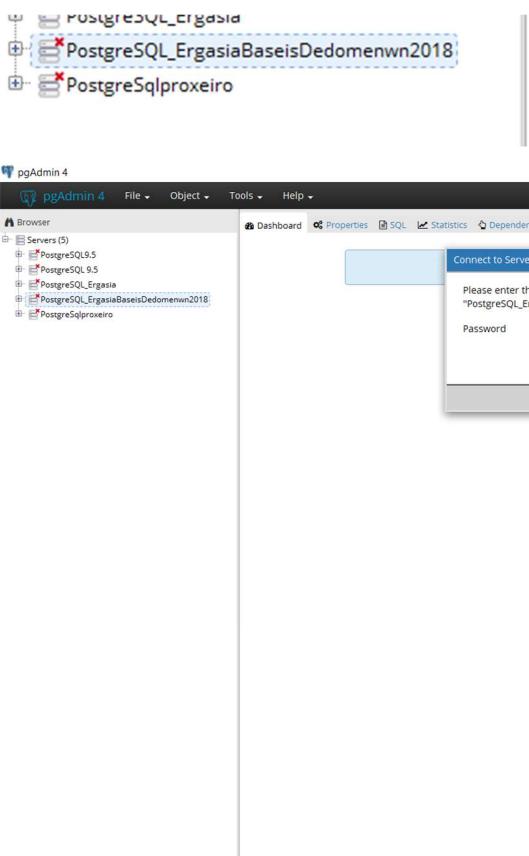


Θα την ονομασω

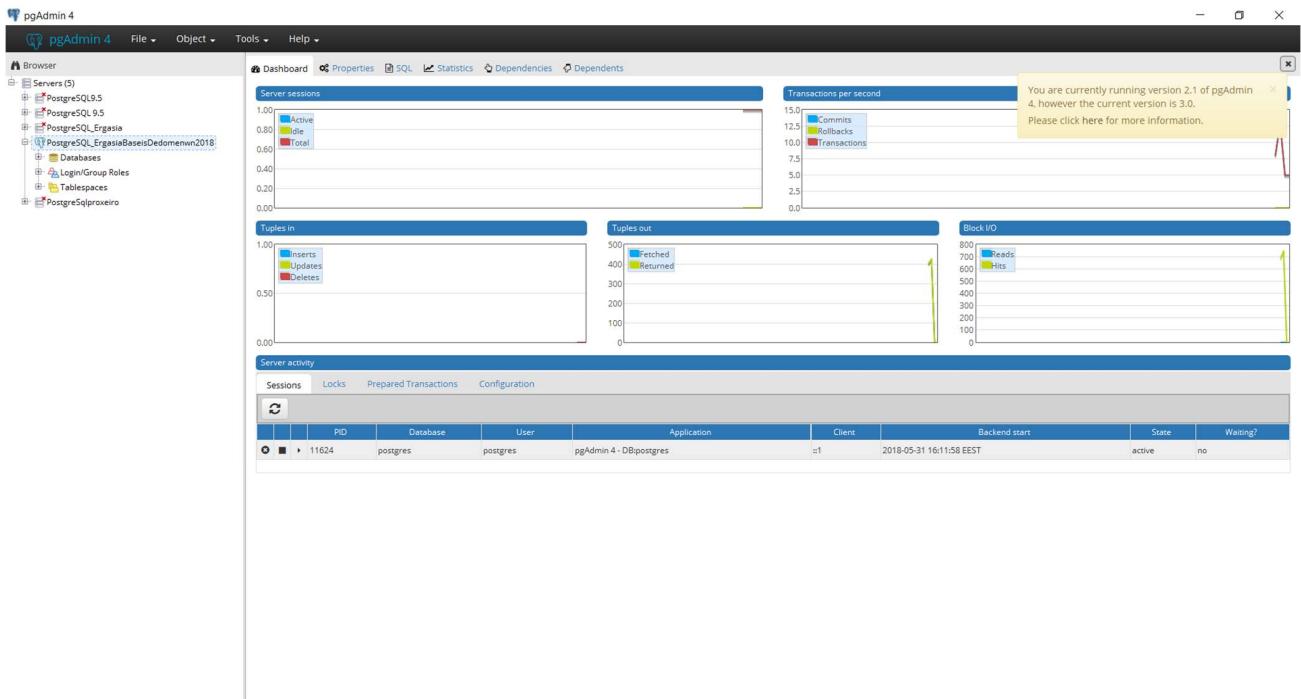


PostgreSQL_ErgasiaBaseisDedomenwn2018

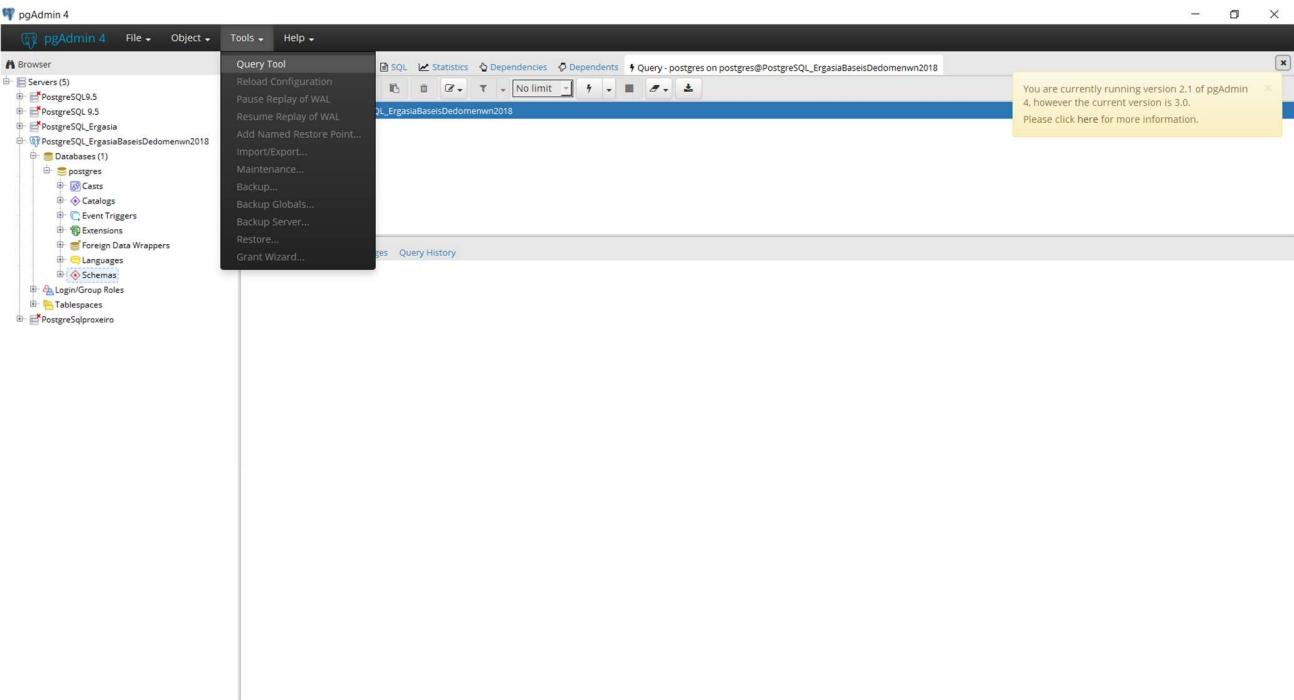
και θα την κανω save



Την δημιουργώ και συνδεομαι πανω της



Συνδεομαι στην βαση δεδομενων μου και πηγαινω πανω στα tools για να ανοιξω ενα query



Εχω φτιαξει τους πινακες μου σε ενα query

```

12
13  create table Company (
14      AFM CHAR(8),
15      Fullname VARCHAR(25),
16      Email varchar(50),
17      primary key (AFM)
18  );
19

```

Φτιαχνω εναν πινακα με ονομα company και χαρακτηριστικα AFM με εκριβες μεγεθος 8 ,Fullname μεχρι 25 χαρακτηρες ,email 50 χαρακτηρες και πρωτευον κλειδι AFM

Εναν πινακα repairs με id_tech με αυστηρο μεγεθος 4 εχω ενα παραδειγμα με σχολιο T001 τεχνικος 1 cost_repair int ακεραιος ,date_repair_start date ημερομηνια πρωτευον κλειδι id_car,date_repair_starts συνδιασμος

```

create table Repairs(
    id_tech char(4),--T001
    id_car char(6),--S18001
    cost_repair int,
    date_repair_starts date,
    primary key (id_car,date_repair_starts)
);

```

Μετα φτιαχνω εναν πινακα Date_ends με id_car μεγεθος αυτοκινητου σχολιο s18001 αυστηρο μεγεθος 6,date_repair_starts date ημερομηνια date_repair_ends date ημερομηνια πρωτευον κλειδι (id_car, date_repair_starts) και βαζω ενα ξενο κλειδι (id_car, date_repair_starts) αναφορα στους repairs

```

27 | create table Date_ends(
28 |   id_car char(6),--S18001
29 |   date_repair_starts date,
30 |   date_repair_ends date,
31 |   primary key (id_car,date_repair_starts),
32 |   foreign key(id_car,date_repair_starts)references Repairs
33 | );
34 |

```

Δημιουργια πινακα Cars με id_car σχολιο coo1 αυστηρο μεγεθος 5,model μεχρι 45 χαρακτηρες creation_date date ημερομηνια ,car_type_used_or_new μεχρι 35 χαρακτηρες πρωτευον κλειδι id_car

```

34 |
35 | create table Cars(
36 |   id_car char(5),--C001
37 |   Model varchar(45),
38 |   creation_date date,
39 |   car_type_used_or_new varchar(35),--newnotsolded newselled used
40 |   primary key (id_car)
41 |
42 | );
43 | . . . . .

```

Δημιουργω εναν πινακα Sells με id_seller αυστηρο μεγεθος 4,id_car αυστηρο μεγεθος 5 ,τα σχολια ειναι παραδειγματα ,car_value int ακεραιος ,car_sold int ακεραιος ,car_transition_date date ημερομηνια εχω βαλει πρωτευον κλειδο το id_car και εξωτερικο κλειδι id_car στον πινακα cars

```

43 | create table Sells(
44 |   id_seller char(4),--S001
45 |   id_car char(5),--C0001
46 |   car_value int,
47 |   car_sold int,
48 |   car_transition_date date,
49 |   primary key(id_car),
50 |   foreign key (id_car)references cars
51 |
52 |

```

Φτιαχνω τον πινακα Technicians με αφμ αυστηρο μεγεθος 8 ,id_tech αυστηρο μεγεθος 4 με πρωτευον κλειδι id_tech και ξενο κλειδι AFM με αναφορα στον πινακα company

```

52
53  create table Technicians(
54      AFM CHAR(8),
55      id_tech char(4),--T001
56      primary key(AFM,id_tech),
57      foreign key (AFM) references Company
58  );
59

```

Φτιαχνω εναν πινακα Sellers με AFM ,id_seller με πρωτευον κλειδι AFM ,id_seller και ξενα κλειδια AFM στον πινακα company

```

59
60  create table Sellers(
61      AFM CHAR(8),
62      id_seller char(4),--S001
63      primary key(AFM,id_seller),
64      foreign key (AFM) references company
65  );
66  create table Customers/

```

Φτιαχνω εναν πινακα customers με afm,phone_number πρωτευον κλειδι afm και ξενο κλειδι afm company

```

66  create table Customers(
67      AFM CHAR(8),
68      phone_number bigint,
69      primary key(AFM),
70      foreign key (AFM) references company
71  );
72

```

Φτιαχνω εναν πινακα Extractcarinfo με model μεγεθος 45 manufacturer μεγεθος χαρακτηρων 30 και πρωτευον κλειδι Model

```

72
73  create table Extractcarinfo(
74      Model varchar(45),
75      Manufacturer varchar(30),
76      primary key (Model)
77  );
78  create table Cartypenew/

```

Φτιαχνω εναν πινακα Cartypenew με id_car αυστηρα 6 ,afm αυστηρα 8 πρωτευον κλειδι id_car,afm ξενα κλειδια id_car στον πινακα και Cars και afm στον πινακα company

```

        ,
        create table Cartypenew(
        id_car char(6),
        AFM CHAR(8),
        primary key(id_car,AFM),
        foreign key (id_car)references Cars,
        foreign key (AFM)references company
);

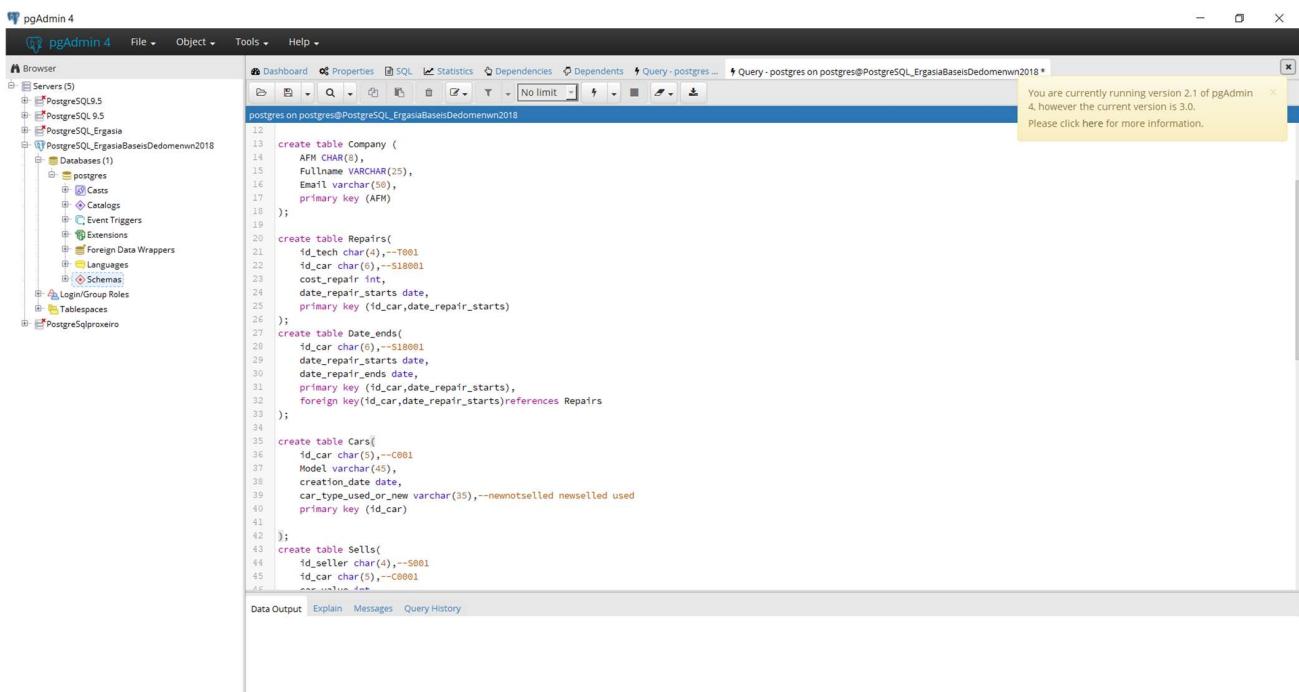
```

Φτιαχνω εναν πινακα Cartypeused με id_car αυστηρα 6 και Licenceplate αυστηρα 6 και πρωτευον κλειδι id_car,Licenceplate και εξωτερικο κλειδι id_car με αναφορα στο Cars

```

85  create table CartypeUsed(
86      id_car char(6),
87      Licenceplate char(8),
88      primary key(id_car,Licenceplate),
89      foreign key (id_car)references Cars
90  );

```



The screenshot shows the pgAdmin 4 interface with the following details:

- Servers:** PostgreSQL 9.5, PostgreSQL 9.5, PostgreSQL_Ergasia, PostgreSQL_ErgasiaBasisDedomenwn2018.
- Databases:** postgres (selected), containing:

 - Tables:** Sells, Technicians, Sellers, Customers, Extracarinfo.
 - Views:** None.
 - Functions:** None.

- Script:** The right pane displays the SQL code for creating these tables. The code is as follows:

```
43 create table Sells(
44     id_seller char(4),--S001
45     id_car char(5),--C001
46     car_value int,
47     car_sold int,
48     car_transition_date date,
49     primary key(id_car),
50     foreign key (id_car) references cars
51 );
52
53 create table Technicians(
54     AFM CHAR(8),
55     id_tech char(4),--T001
56     primary key(AFM,id_tech),
57     foreign key (AFM) references Company
58 );
59
60 create table Sellers(
61     AFM CHAR(8),
62     id_seller char(4),--S001
63     primary key(AFM,id_seller),
64     foreign key (AFM) references company
65 );
66 create table Customers(
67     AFM CHAR(8),
68     phone_number bigint,
69     primary key(AFM),
70     foreign key (AFM) references company
71 );
72
73 create table Extracarinfo(
74     Model varchar(45),
75     Manufacturer varchar(30),
76     primary key (Model)
```

- Status Bar:** You are currently running version 2.1 of pgAdmin 4, however the current version is 3.0. Please click here for more information.

```

create table Extracarinfo(
    Model varchar(45),
    Manufacturer varchar(30),
    primary key (Model)
);
create table Cartypenew(
    id_car char(6),
    AFM CHAR(8),
    primary key(id_car,AFM),
    foreign key (id_car)references Cars,
    foreign key (AFM)references company
);
create table CartypeUsed(
    id_car char(6),
    Licenceplate char(8),
    primary key(id_car,Licenceplate),
    foreign key (id_car)references Cars
);

```

ta Output Explain Messages Query History

Στην συνχεια θελω να περασω τα δεδομενα μου απο τα excel αρχεια που εδειξα ποιο πανω στην postgresql.

Βασεις Δεδομένων p14086 , p14036

Σελίδα 26 από 67

The screenshot shows the pgAdmin 4 interface. On the left, the 'Browser' panel displays a tree view of servers and databases. Under 'Servers (5)', 'PostgreSQL_ErgasiaBaseisDedomenwn2018' is selected, showing its 'Databases (1)' containing 'postgres'. Inside 'postgres', various objects like 'Casts', 'Catalogs', 'Event Triggers', 'Extensions', 'Foreign Data Wrappers', 'Languages', and 'Schemas' are listed. On the right, the main window has a tab bar with 'Dashboard', 'Properties', 'SQL', 'Statistics', 'Dependencies', 'Dependents', 'Query - postgres ...', and 'Query...'. The 'Query...' tab is active, showing a SQL query window with the following content:

```
1 copy company from 'D:\Company.csv' Delimiter ','CSV HEADER;
2 copy technicians from 'D:\Technicians.csv' Delimiter ','CSV HEADER;
3 copy sellers from 'D:\Sellers.csv' Delimiter ','CSV HEADER;
4 copy customers from 'D:\Customers.csv' Delimiter ','CSV HEADER;
5 copy repairs from 'D:\Repairs.csv' Delimiter ','CSV HEADER;
6 copy Date_ends from 'D:\Date_ends.csv' Delimiter ','CSV HEADER;
7 copy cars from 'D:\Cars.csv' Delimiter ','CSV HEADER;
8 copy sells from 'D:\Sells.csv' Delimiter ','CSV HEADER;
9 copy extracarinfo from 'D:\Extracarinfo.csv' Delimiter ','CSV HEADER;
10 copy cartypenew from 'D:\Cartypenew.csv' Delimiter ','CSV HEADER;
11 copy cartypeused from 'D:\Cartypeused.csv' Delimiter ','CSV HEADER;
12
```

copy company from 'D:\Company.csv' Delimiter ','CSV HEADER;

Κανω αντιγραφη στον πινακα company απο το αντιστοιχο path που εχω τα δεδομενα μου εδω τα εχω ονομασει οπως οι πινακες μου απο τα csv αρχεια

The screenshot shows the pgAdmin 4 interface. On the left, the 'Browser' panel displays a tree view of database objects. Under 'Servers (5)', 'PostgreSQL_ErgasiaBaseisDedomenwn2018' is selected, showing its 'Databases (1)' containing 'postgres'. Inside 'postgres', there are 'Schemas (1)' with 'public' selected, showing various table definitions like 'Cars', 'Customers', etc. On the right, the main window shows a SQL query window with the following code:

```

1 copy company from 'D:\Company.csv' Delimiter ','CSV HEADER;
2 copy technicians from 'D:\Technicians.csv' Delimiter ','CSV HEADER;
3 copy sellers from 'D:\Sellers.csv' Delimiter ','CSV HEADER;
4 copy customers from 'D:\Customers.csv' Delimiter ','CSV HEADER;
5 copy repairs from 'D:\Repairs.csv' Delimiter ','CSV HEADER;
6 copy Date_ends from 'D:\Date_ends.csv' Delimiter ','CSV HEADER;
7 copy cars from 'D:\Cars.csv' Delimiter ','CSV HEADER;
8 copy sells from 'D:\Sells.csv' Delimiter ','CSV HEADER;
9 copy extracarinfo from 'D:\Extracarinfo.csv' Delimiter ','CSV HEADER;
10 copy cartypenew from 'D:\Cartypenew.csv' Delimiter ','CSV HEADER;
11 copy cartypeused from 'D:\Cartypeused.csv' Delimiter ','CSV HEADER;

```

The 'Data Output' tab is active at the bottom.

Βλεπουμε τους πινακες που δημιουργησαμε

Παταω δεξι click view all rows σε καθε πινακα

pgAdmin 4

pgAdmin 4 File Object Tools Help

Browser

- Servers (5)
 - PostgreSQL9.5
 - PostgreSQL 9.5
 - PostgreSQL_Ergasia
 - PostgreSQL_ErgasiaBaseisDedomenwn2018
 - Databases (1)
 - postgres
 - Casts
 - Catalogs
 - Event Triggers
 - Extensions
 - Foreign Data Wrappers
 - Languages
 - Schemas (1)
 - public
 - Collations
 - Domains
 - FTS Configurations
 - FTS Dictionaries
 - FTS Parsers
 - FTS Templates
 - Foreign Tables
 - Functions
 - Materialized Views
 - Sequences (1)
 - Tables (11)
 - cars
 - cartypenew
 - cartypeused
 - company
 - customers
 - date_ends
 - extracarinfo
 - repairs
 - sellers
 - sells
 - technicians
 - Trigger Functions
 - Types
 - Views

PostgreSQL_ErgasiaBaseisDedomenwn2018 - postgres - public.cars

```

1 SELECT * FROM public.cars
2 ORDER BY id_car ASC

```

Data Output Explain Messages Query History

	id_car [PK] character (5)	model character varying (45)	creation_date date	car_type_used_or_new character varying (35)
1	C0000	Cirrus	2009-09-27	newselled
2	C0001	Mulsanne	2009-02-05	newselled
3	C0002	4000s Quattro	2009-05-07	newselled
4	C0003	Sonata	2009-09-08	used
5	C0004	M6	2009-03-23	newselled
6	C0005	Tempest	2009-02-17	newselled
7	C0006	tC	2009-09-03	newselled
8	C0007	Expo	2009-12-03	newselled
9	C0008	Tahoe	2009-05-28	newselled
10	C0009	Durango	2009-12-17	used
11	C0010	Bonneville	2009-09-17	newselled
12	C0011	Montana	2009-11-08	newselled
13	C0012	Cobalt	2009-07-24	newselled
14	C0013	Camaro	2009-07-13	used
15	C0014	Seville	2009-11-06	newselled
16	C0015	XJ Series	2009-06-12	newselled
17	C0016	GL-Class	2009-06-16	newselled
18	C0017	Enclave	2009-04-24	newselled
19	C0018	Hombre Space	2009-04-17	used
20	C0019	HHR	2009-06-03	newselled
21	C0020	Ram 1500	2009-04-19	newselled
22	C0021	Caliber	2009-09-11	used
23	C0022	98	2009-11-07	newselled
24	C0023	Familia	2009-02-26	newselled

πινακας cars

cartypenew

pgAdmin 4

pgAdmin 4 File Object Tools Help

Browser

- Servers (5)
 - PostgreSQL9.5
 - PostgreSQL 9.5
 - PostgreSQL_Ergasia
 - PostgreSQL_ErgasiaBaseisDedomenwn2018
 - Databases (1)
 - postgres
 - Casts
 - Catalogs
 - Event Triggers
 - Extensions
 - Foreign Data Wrappers
 - Languages
 - Schemas (1)
 - public
 - Collations
 - Domains
 - FTS Configurations
 - FTS Dictionaries
 - FTS Parsers
 - FTS Templates
 - Foreign Tables
 - Functions
 - Materialized Views
 - Sequences
 - Tables (11)
 - cars
 - cartypenew
 - cartypeused
 - company
 - customers
 - date_ends
 - extracarinfo
 - repairs
 - sellers
 - sells
 - technicians
 - Trigger Functions
 - Types
 - Views

Dashboard Properties SQL Statistics Dependencies

PostgreSQL_ErgasiaBaseisDedomenwn2018 - postgres - public.cartypenew

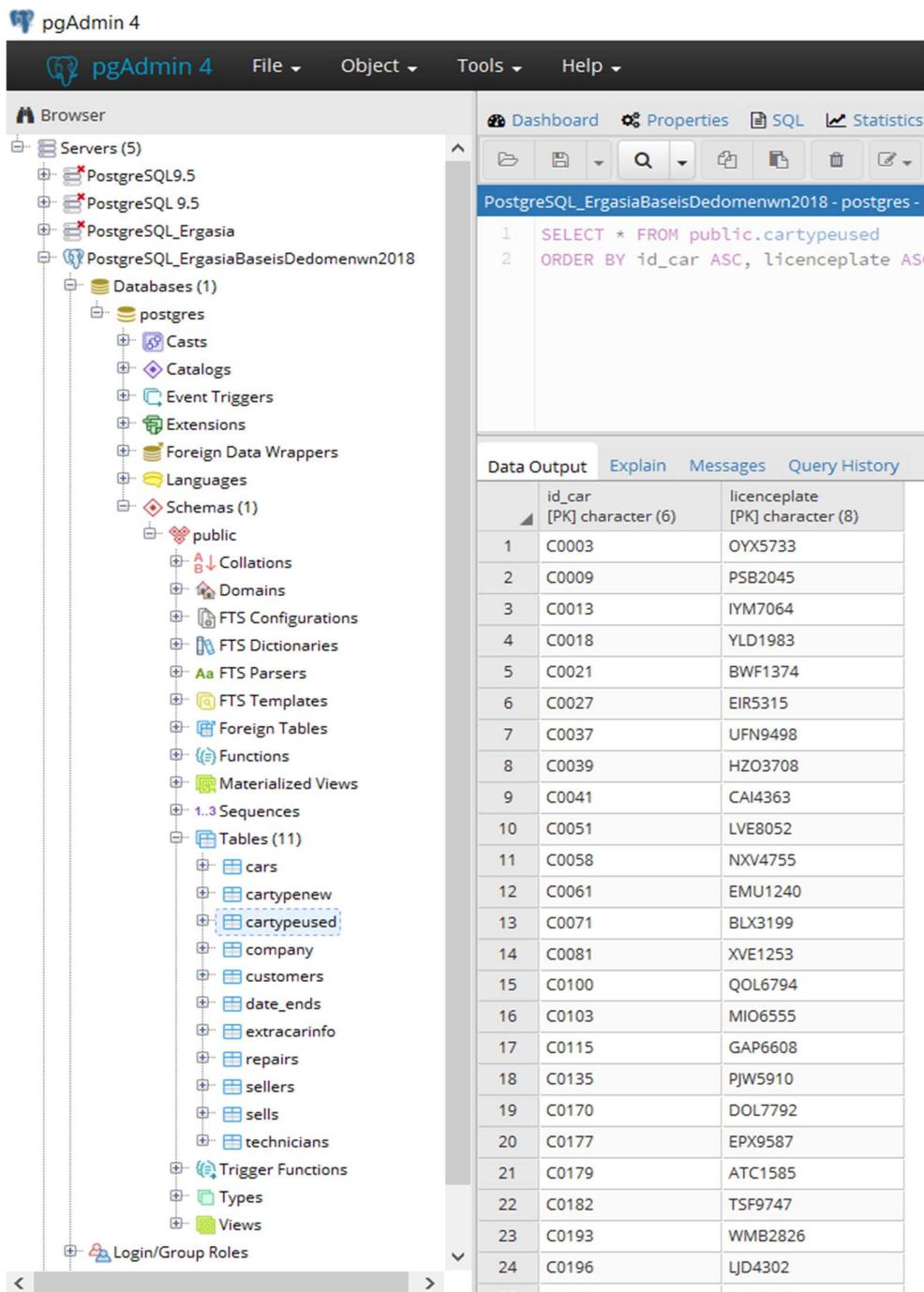
```

1 SELECT * FROM public.cartypenew
2 ORDER BY id_car ASC, afm ASC

```

	id_car [PK] character (6)	afm [PK] character (8)
1	C0000	53315050
2	C0001	12530189
3	C0002	51748644
4	C0004	93276961
5	C0005	76929065
6	C0006	90050585
7	C0007	66571260
8	C0008	94891468
9	C0010	95851113
10	C0011	67414367
11	C0012	52504726
12	C0014	74656951
13	C0015	92018458
14	C0016	68680761
15	C0017	26232015
16	C0019	28083395
17	C0020	66186185
18	C0022	48817323
19	C0023	80931880
20	C0024	83045092
21	C0025	96402874
22	C0026	10758422
23	C0028	56534685
24	C0029	83665305

cartypeused



pgAdmin 4

pgAdmin 4 File Object Tools Help

Browser

- Servers (5)
 - PostgreSQL9.5
 - PostgreSQL 9.5
 - PostgreSQL_Ergasia
 - PostgreSQL_ErgasiaBaseisDedomenwn2018
 - Databases (1)
 - postgres
 - Casts
 - Catalogs
 - Event Triggers
 - Extensions
 - Foreign Data Wrappers
 - Languages
 - Schemas (1)
 - public
 - Collations
 - Domains
 - FTS Configurations
 - FTS Dictionaries
 - FTS Parsers
 - FTS Templates
 - Foreign Tables
 - Functions
 - Materialized Views
 - Sequences
 - Tables (11)
 - cars
 - cartypernew
 - cartyused
 - company
 - customers
 - date_ends
 - extracarinfo
 - repairs
 - sellers
 - sells
 - technicians
 - Trigger Functions
 - Types
 - Views
 - Login/Group Roles

Dashboard Properties SQL Statistics Dependencies Dependencies No limit

PostgreSQL_ErgasiaBaseisDedomenwn2018 - postgres - public.company

```
1 SELECT * FROM public.company
2 ORDER BY afm ASC
```

	afm [PK] character (8)	fullname character varying (25)	email character varying (50)
1	10435649	Courtenay Artus	cartusq2@taobao.com
2	10497325	Neville Dudgeon	ndudgeonfk@prlog.org
3	10554644	Marysa Burkart	mburkartaq@earthlink.net
4	10572930	Silva Kew	skew3l@mapquest.com
5	106303	Hiram Bauldry	hbauldreyu@berkeley.edu
6	10647974	Barbi Tapley	btapleyls@t.co
7	10758422	Garrard Ledamun	gledamunq@europa.eu
8	10829229	Desirae Charnick	dcharnickem@rediff.com
9	10892737	Forest Attard	fattardig@statcounter.com
10	11008308	Paddie Gulliford	pgullifordmj@dmoz.org
11	11012800	Carina Krzyzaniak	ckrzyzaniaklt@nymag.com
12	11256994	Ondrea Matashkin	omatashkinn1@sitemeter....
13	11277750	Ursola Nickell	unicellhct@posterous.com
14	11334275	Johann Yeude	jyeudea5@desdev.cn
15	11381939	Pansy Filippucci	pfilippucci8j@opera.com
16	11383980	Christean Chavrin	cchavrinfid@ow.ly
17	11479737	Martita Allcoat	mallcoat1@tamu.edu
18	11570735	Gilberte Boschers	gbosherspq@google.nl
19	11591566	Terriann Thickin	tthickinmg@elpais.com
20	11627142	Tobe Pointing	tpointingl9@purevolume.c...
21	11651509	Zechariah Lille	zlille5l@theguardian.com
22	11659966	Lionello Mannerling	lmanneringik@sphinn.com
23	11730389	Hastings Taggart	htaggart9f@yale.edu
24	11757528	Rickert Bryce	rbryce3r@behance.net

company

Customer

pgAdmin 4

pgAdmin 4 File Object Tools Help

Browser

- Servers (5)
 - PostgreSQL9.5
 - PostgreSQL 9.5
 - PostgreSQL_Ergasia
 - PostgreSQL_ErgasiaBaseisDedomenwn2018
 - Databases (1)
 - postgres
 - Casts
 - Catalogs
 - Event Triggers
 - Extensions
 - Foreign Data Wrappers
 - Languages
 - Schemas (1)
 - public
 - Collations
 - Domains
 - FTS Configurations
 - FTS Dictionaries
 - FTS Parsers
 - FTS Templates
 - Foreign Tables
 - Functions
 - Materialized Views
 - Sequences
 - Tables (11)
 - cars
 - cartypenew
 - cartypeused
 - company
 - customers
 - date_ends
 - extracarinfo
 - repairs
 - sellers
 - sells
 - technicians
 - Trigger Functions
 - Types
 - Views
 - Login/Group Roles

Dashboard Properties SQL Statist

PostgreSQL_ErgasiaBaseisDedomenwn2018 - postgres

```

1 SELECT * FROM public.customers
2 ORDER BY afm ASC
  
```

	afm [PK] character (8)	phone_number bigint
1	10435649	3253324655
2	10497325	6775899979
3	10554644	9522769830
4	10572930	4585222471
5	106303	2283682956
6	10647974	8739302191
7	10758422	3177446190
8	10829229	4759364056
9	10892737	7792358364
10	11012800	7866461075
11	11256994	5033648175
12	11277750	9032098742
13	11334275	5621732184
14	11381939	1509051565
15	11383980	8181574498
16	11479737	7824773222
17	11570735	4744102720
18	11627142	3055864619
19	11651509	7137938828
20	11659966	8997092551
21	11730389	9802307460
22	11757528	2479127883
23	11789956	7011631456
24	12030439	2474947784
25	12041050	2474947784

date_ends

The screenshot shows the pgAdmin 4 interface. In the left-hand sidebar, under 'Servers (5)', the 'PostgreSQL_ErgasiaBaseisDedomenwn2018' server is selected. Under 'Databases (1)', the 'postgres' database is selected. Inside 'Tables (11)', the 'date_ends' table is highlighted. The main pane displays the SQL query:

```
1 SELECT * FROM public.date_ends
2 ORDER BY id_car ASC, date_repair_starts ASC
```

The 'Data Output' tab shows the results of the query:

	id_car [PK] character (6)	date_repair_starts [PK] date	date_repair_ends date
1	C0000	2016-10-31	2016-11-06
2	C0000	2017-05-07	2017-05-14
3	C0001	2017-05-10	2017-05-12
4	C0001	2017-09-20	2017-09-24
5	C0002	2011-01-12	2011-01-16
6	C0003	2012-03-03	2012-03-04
7	C0003	2012-07-08	2012-07-18
8	C0003	2014-11-30	2014-12-10
9	C0004	2011-01-30	2011-02-03
10	C0005	2011-02-10	2011-02-14
11	C0005	2012-12-26	2012-12-31
12	C0005	2013-04-03	2013-04-08
13	C0006	2013-12-26	2013-12-30
14	C0006	2016-04-09	2016-04-11
15	C0006	2017-02-20	2017-02-27
16	C0007	2014-02-26	2014-03-02
17	C0007	2014-03-07	2014-03-10
18	C0007	2017-08-17	2017-08-18
19	C0007	2018-03-28	2018-04-08
20	C0008	2013-01-28	2013-02-01
21	C0010	2011-02-03	2011-02-11
22	C0010	2016-08-27	2016-09-04
23	C0011	2011-01-18	2011-01-24
24	C0011	2012-05-03	2012-05-13

extrcarinfo

pgAdmin 4

pgAdmin 4 File Object Tools Help

Browser

- Servers (5)
 - PostgreSQL9.5
 - PostgreSQL 9.5
 - PostgreSQL_Ergasia
 - PostgreSQL_ErgasiaBaseisDedomenwn2018
- Databases (1)
 - postgres
 - Casts
 - Catalogs
 - Event Triggers
 - Extensions
 - Foreign Data Wrappers
 - Languages
 - Schemas (1)
 - public
 - Collations
 - Domains
 - FTS Configurations
 - FTS Dictionaries
 - FTS Parsers
 - FTS Templates
 - Foreign Tables
 - Functions
 - Materialized Views
 - Sequences
 - Tables (11)
 - cars
 - cartypenew
 - cartypeused
 - company
 - customers
 - date_ends
 - extracarinfo
 - repairs
 - sellers
 - sells
 - technicians
 - Trigger Functions
 - Types
 - Views
- Login/Group Roles

Dashboard Properties SQL Statistics Dependencies

PostgreSQL_ErgasiaBaseisDedomenwn2018 - postgres - public.extracarinfo

```
1 SELECT * FROM public.extracarinfo
2 ORDER BY model ASC
```

Data Output Explain Messages Query History

	model [PK] character varying (45)	manufacturer character varying (30)
1	100	Quamba
2	1000	Omiba
3	1500	Trilia
4	1500 Club Coupe	Mydo
5	164	Thoughtstorm
6	200	Yoveo
7	228	Skinix
8	240	Centimia
9	2500	Talane
10	3-Sep	Realcube
11	3 Series	Ooba
12	300D	Yadel
13	300M	Browselab
14	300SE	Yakitri
15	300ZX	Twimm
16	323	Edgepulse
17	325	Fiveclub
18	3500	Babblestorm
19	3500 Club Coupe	Vimbo
20	350Z	Thoughtworks
21	4000s	Blogtag
22	4000s Quattro	Podcat
23	400E	InnoZ
24	4Runner	Myworks

repairs

The screenshot shows the pgAdmin 4 interface. On the left, the 'Browser' pane displays the database structure under 'PostgreSQL_ErgasiaBaseisDedomenwn2018'. Under 'Tables', there is a table named 'repairs'. On the right, the main window shows a query results grid for the 'repairs' table.

Query results:

```
1 SELECT * FROM public.repairs
2 ORDER BY id_car ASC, date_repair_starts ASC
```

	id_tech	id_car	cost_repair	date_repair_starts
1	T033	C0000	1643	2016-10-31
2	T032	C0000	446	2017-05-07
3	T027	C0001	833	2017-05-10
4	T085	C0001	401	2017-09-20
5	T092	C0002	390	2011-01-12
6	T023	C0003	1913	2012-03-03
7	T017	C0003	245	2012-07-08
8	T071	C0003	549	2014-11-30
9	T063	C0004	210	2011-01-30
10	T082	C0005	1037	2011-02-10
11	T063	C0005	1941	2012-12-26
12	T071	C0005	1152	2013-04-03
13	T033	C0006	623	2013-12-26
14	T081	C0006	677	2016-04-09
15	T037	C0006	1608	2017-02-20
16	T045	C0007	983	2014-02-26
17	T093	C0007	762	2014-03-07
18	T000	C0007	1618	2017-08-17
19	T087	C0007	933	2018-03-28
20	T061	C0008	887	2013-01-28
21	T041	C0010	470	2011-02-03
22	T052	C0010	847	2016-08-27
23	T078	C0011	836	2011-01-18
24	T082	C0011	738	2012-05-03

sellers

pgAdmin 4

pgAdmin 4 File Object Tools Help

Browser

- Servers (5)
 - PostgreSQL9.5
 - PostgreSQL 9.5
 - PostgreSQL_Ergasia
 - PostgreSQL_ErgasiaBaseisDedomenwn2018
- Databases (1)
 - postgres
 - Casts
 - Catalogs
 - Event Triggers
 - Extensions
 - Foreign Data Wrappers
 - Languages
 - Schemas (1)
 - public
 - Collations
 - Domains
 - FTS Configurations
 - FTS Dictionaries
 - FTS Parsers
 - FTS Templates
 - Foreign Tables
 - Functions
 - Materialized Views
 - Sequences
 - Tables (11)
 - cars
 - cartypernew
 - cartypeused
 - company
 - customers
 - date_ends
 - extracarinfo
 - repairs
 - sellers
 - sells
 - technicians
 - Trigger Functions
 - Types
 - Views
 - Login/Group Roles

Dashboard Properties SQL Statistics

PostgreSQL_ErgasiaBaseisDedomenwn2018 - postgres

```

1 SELECT * FROM public.sellers
2 ORDER BY afm ASC, id_seller ASC

```

	afm [PK] character (8)	id_seller [PK] character (4)
1	10892737	S016
2	11659966	S056
3	12040358	S009
4	15139792	S044
5	15146069	S025
6	17104536	S011
7	17412063	S078
8	17607073	S042
9	20526140	S020
10	2226650	S030
11	22748213	S021
12	23198293	S071
13	23303095	S053
14	2566375	S097
15	26699559	S029
16	28104565	S026
17	28335984	S003
18	29342195	S049
19	29703226	S087
20	30348538	S066
21	30597250	S010
22	31981508	S070
23	32303073	S035
24	32557858	S072
...

sells

pgAdmin 4

pgAdmin 4 File Object Tools Help

Browser

- Servers (5)
 - PostgreSQL9.5
 - PostgreSQL 9.5
 - PostgreSQL_Ergasia
 - PostgreSQL_ErgasiaBaseisDedomenwn2018
 - Databases (1)
 - postgres
 - Casts
 - Catalogs
 - Event Triggers
 - Extensions
 - Foreign Data Wrappers
 - Languages
 - Schemas (1)
 - public
 - Collations
 - Domains
 - FTS Configurations
 - FTS Dictionaries
 - FTS Parsers
 - FTS Templates
 - Foreign Tables
 - Functions
 - Materialized Views
 - Sequences
 - Tables (11)
 - cars
 - cartypenew
 - cartypeused
 - company
 - customers
 - date_ends
 - extracarinfo
 - repairs
 - sellers
 - sells
 - technicians
 - Trigger Functions
 - Types
 - Views
 - Login/Group Roles

Dashboard Properties SQL Statistics Dependencies Dependents Query

PostgreSQL_ErgasiaBaseisDedomenwn2018 - postgres - public.sells

```
1 SELECT * FROM public.sells
2 ORDER BY id_car ASC
```

Data Output Explain Messages Query History

	id_seller character (4)	id_car [PK] character (5)	car_value integer	car_sold integer	car_transition_date date
1	S097	C0000	14976	15576	2010-11-01
2	S054	C0001	11133	11733	2010-03-12
3	S078	C0002	12803	13403	2010-06-11
4	S029	C0003	16014	16614	2010-10-13
5	S047	C0004	17274	17874	2010-04-27
6	S099	C0005	16225	16825	2010-03-24
7	S020	C0006	10845	11445	2010-10-08
8	S093	C0007	11375	11975	2011-01-07
9	S045	C0008	10960	11160	2010-07-02
10	S023	C0009	18616	18816	2011-01-21
11	S033	C0010	19999	20199	2010-10-22
12	S039	C0011	11185	11385	2010-12-13
13	S015	C0012	11407	11607	2010-08-28
14	S061	C0013	17571	17771	2010-08-17
15	S051	C0014	15058	15258	2010-12-11
16	S079	C0015	13868	14068	2010-07-17
17	S036	C0016	14518	14718	2010-07-21
18	S082	C0017	13380	13580	2010-05-29
19	S010	C0018	16309	16509	2010-05-22
20	S089	C0019	15107	15307	2010-07-08
21	S042	C0020	18071	18271	2010-05-24
22	S063	C0021	18005	18205	2010-10-16
23	S006	C0022	14412	14612	2010-12-12
24	S074	C0023	17700	17900	2010-04-02
25	S056	C0024	16205	16205	2010-07-10

technicians

The screenshot shows the pgAdmin 4 interface. On the left, the tree view displays the database structure under 'PostgreSQL_ErgasiaBaseisDedomenwn2018 - postgres'. The 'Tables' node is expanded, showing 11 tables: cars, cartypenew, cartypeused, company, customers, date_ends, extracarinfo, repairs, sellers, sells, and technicians. The 'Technicians' table is highlighted. On the right, the main window shows a SQL query in the 'Query' tab:

```
1 SELECT * FROM public.technicians
2 ORDER BY afm ASC, id_tech ASC
```

The results are displayed in a table titled 'Data Output' with 25 rows. The columns are 'afm [PK] character (8)' and 'id_tech [PK] character (4)'. The data is as follows:

	afm [PK] character (8)	id_tech [PK] character (4)
1	11381939	T060
2	11730389	T092
3	13297823	T023
4	15542047	T052
5	15595440	T051
6	15735401	T069
7	16083821	T006
8	16117999	T079
9	19011435	T024
10	19519651	T077
11	20234496	T059
12	21536939	T042
13	23092522	T066
14	2325769	T056
15	23269937	T007
16	23677913	T014
17	24027257	T010
18	24054358	T031
19	24642862	T084
20	28090311	T004
21	3008783	T074
22	30449501	T086
23	31338058	T001
24	31888089	T039
25	32151607	T005

(b) Εφαρμόστε τη θεωρία της κανονικοποίησης πάνω στο σχεσιακό σχήμα της ΒΔ που σχεδιάσατε και ελέγξτε εάν ακολουθεί την 3NF ή BCNF. Τώρα θα εξετασω αν το σχήμα μου ειναι σε μορφή 3nf

1)Τα πεδια των τιμων ολων των χαρακτηριστικων ειναι ατομικα 1NF

2) Κανενα χαρακτηριστικο απο αυτα που δεν ειναι κλειδια δεν προσδιοριζεται απο τμημα ενος υποψηφιου κλειδιου 2NF

3) Κανενα χαρακτηριστικο απο αυτα που δεν συμμετεχουν σε υποψηφιο κλειδι δεν προσδιοριζεται μεταβατικα (μεσω άλλου χαρακτηριστικου - το οποιο δεν συμμετέχει σε υποψηφιο κλειδι) απο ενα υποψηφιο κλειδι.

Company: AFM → Fullname, Email

Technicians: AFM, id_tech

Sellers: AFM, id_seller

Customers: AFM → phone_number

Repairs: id_car, date_repair_starts → id_tech, cost_repair

Date_ends: id_car, date_repair_starts → date_repair_ends

Sells: id_car → id_seller, car_value, car_sold, car_transition_date

Cars: id_car → Model, creation_date, car_type_used-or-new

Extracarinfo: Model → Manufacturer

CartypeUsed: id_car → Licenceplate

Cartypenew : id_car, AFM

Αρα καταλαβαινουμε οτι ο Πινακας εινα σε μορφή 3NF

Ερώτημα 2 (30%).

Εκτελέστε τις παρακάτω ερωτήσεις (queries) στη ΒΔ (εντολές SELECT).

(a) Ποια είναι τα μοντέλα αυτοκινήτων με το μέγιστο πλήθος ζημιών.

Αρχικα ανοιχω ενα query για να γραψω την εντολη

The screenshot shows the pgAdmin 4 interface. On the left, the 'Browser' panel displays a tree view of servers, databases, and schema objects. The current connection is to 'PostgreSQL_ErgasiaBaseisDedomenwn2018'. In the center, the 'Query Editor' window contains the following SQL code:

```
1 select repairs.id_car,sum(cost_repair),cars.model
2 from repairs
3 inner join cars
4 on repairs.id_car=cars.id_car
5 group by repairs.id_car,cars.model
6 having sum(cost_repair) in(
7 select max(sum_cost)
8 from(select id_car,sum(cost_repair)as sum_cost from repairs group by id_car)as temporaryTable)
9
10
11 --http://sqlfiddle.com/#12/38e44/1
```

Below the query editor, the 'Data Output' tab is selected, showing the results of the query:

id_car	sum	model
C0136	12621	Jetta
C0472	12621	Suburban 2500

το σχολιο που εχω βαλει ειναι πηγη που χρησιμοποιησα για να γραψω την εντολη

```
select repairs.id_car,sum(cost_repair),cars.model
from repairs
inner join cars
on repairs.id_car=cars.id_car
group by repairs.id_car,cars.model
having sum(cost_repair) in(
select max(sum_cost)
from(select id_car,sum(cost_repair)as sum_cost from repairs group by id_car)as temporaryTable)
```

Αυτο που κανω ειναι να διαλεξω και να συνδεσω εσωτερικα δυο πινακες των repairs με τον cars να αθροισω το κοστος επισκευης για καθε αυτοκινητο (**group by**)και μετα φτιαχνω εναν προσωρινο πινакα για να βρω τον μεγιστο κοστος επισκευης απο ολα τα αυτοκινητα και ετσι να εμφανισω εκεινα που εχουν το μεγαλυτερο κοστος

Και βρισκω οτι ηταν δυο με μεγιστο κοστος

"C0136 " "12621" "Jetta"
"C0472 " "12621" "Suburban 2500"

(b) Ποιο είναι το μέσο κέρδος της εταιρείας από επισκευές (= μέσο κόστος ζημιών) ανά μήνα.

pgAdmin 4

The screenshot shows the pgAdmin 4 interface. On the left, the Browser pane displays the database structure under 'PostgreSQL_ErgasiaBaseisDedomenwn2018'. In the center, the main window has a toolbar at the top with various icons. Below the toolbar, the title bar says 'postgres on postgres@PostgreSQL_ErgasiaBaseisDedomenwn2018'. The SQL tab contains the following query:

```

1 select date_trunc('month',date_repair_starts),avg(cost_repair)as Avg_earnings_bymonth
from repairs
group by date_trunc('month', date_repair_starts)
order by date_trunc('month', date_repair_starts)

```

The Data Output tab shows the results of the query:

	date_trunc timestamp with time zone	avg_earnings_bymonth numeric
1	2010-05-01 00:00:00+03	830.2500000000000000
2	2010-06-01 00:00:00+03	1216.9285714285714286
3	2010-07-01 00:00:00+03	1099.9615384615384615
4	2010-08-01 00:00:00+03	1178.7307692307692308
5	2010-09-01 00:00:00+03	1303.8666666666666667
6	2010-10-01 00:00:00+03	920.2727272727272727
7	2010-11-01 00:00:00+02	1011.2142857142857143
8	2010-12-01 00:00:00+02	1072.8444444444444444
9	2011-01-01 00:00:00+02	1076.6415094339622642
10	2011-02-01 00:00:00+02	989.4259259259259259
11	2011-03-01 00:00:00+02	1078.3877551020408163
12	2011-04-01 00:00:00+03	1125.3571428571428571
13	2011-05-01 00:00:00+03	1105.4062500000000000
14	2011-06-01 00:00:00+03	1081.4285714285714286
15	2011-07-01 00:00:00+03	1206.8095238095238095
16	2011-08-01 00:00:00+03	1186.8095238095238095
17	2011-09-01 00:00:00+03	1235.3750000000000000

```

select date_trunc('month',date_repair_starts),avg(cost_repair)as Avg_earnings_bymonth
from repairs
group by date_trunc('month', date_repair_starts)
order by date_trunc('month', date_repair_starts)

```

Κανω select απο τις ημερομηνιες οπου ξεκινανε οι επισκευες τον καθε μηνα και τα κανω group by με βαση τον μην και βρίσκω το μεσο κοστος για καθε μηνα απο τον πινακα repairs

```

"2010-05-01 00:00:00+03" "830.2500000000000000"
"2010-06-01 00:00:00+03" "1216.9285714285714286"
"2010-07-01 00:00:00+03" "1099.9615384615384615"
"2010-08-01 00:00:00+03" "1178.7307692307692308"
"2010-09-01 00:00:00+03" "1303.8666666666666667"
"2010-10-01 00:00:00+03" "920.2727272727272727"
"2010-11-01 00:00:00+02" "1011.2142857142857143"

```

Ενδεικτικα εχω μερικες απαντησεις απο το query που ετρεξα

(c) Ποιος είναι ο πωλητής με το μέγιστο «τζίρο» (= αξία πωλήσεων – αξία αγορών).

The screenshot shows the pgAdmin 4 interface. On the left, the 'Browser' pane displays a tree view of database objects. Under 'Servers', there are three entries: PostgreSQL9.5, PostgreSQL9.5, and PostgreSQL_Ergasia. Under 'PostgreSQL_Ergasia', it shows 'Databases (1)' containing 'postgres', and 'Schemas (1)' containing 'public'. The 'public' schema includes 'Collations', 'Domains', 'FTS Configurations', 'FTS Dictionaries', 'FTS Parsers', 'FTS Templates', 'Foreign Tables', 'Functions', 'Materialized Views', and '1..3 Sequences'. The right pane is titled 'postgres on postgres@PostgreSQL_ErgasiaBaseisDedomenwn2018'. It contains a SQL query window with the following code:

```

1 select sum(car_sold-car_value) ,sells.id_seller,sellers.afm,company.Fullname
2 from sells
3 inner join sellers
4 on sells.id_seller=sellers.id_seller
5 inner join company
6 on sellers.afm=company.afm
7 group by sells.id_seller,sellers.afm,company.Fullname
8 having sum(car_sold-car_value)in(
9 select max(difference)
10 from(select id_seller,sum(car_sold-car_value)as difference  from sells group by id_seller)as temporaryTable)
11

```

Below the query window, there are tabs for 'Data Output', 'Explain', 'Messages', and 'Query History'. The 'Data Output' tab shows a single row of results:

	sum	id_seller	afm	fullname
1	10674	S091	95168474	Aundrea Didsbury

```

select sum(car_sold-car_value) ,sells.id_seller,sellers.afm,company.Fullname
from sells
inner join sellers
on sells.id_seller=sellers.id_seller
inner join company
on sellers.afm=company.afm
group by sells.id_seller,sellers.afm,company.Fullname
having sum(car_sold-car_value)in(
select max(difference)
from(select id_seller,sum(car_sold-car_value)as difference from sells group by id_seller)as
temporaryTable)

```

Διαλεγω ομαδοποιημενα με βαση τους πωλητες το αθροισμα της διαφορας πωληση – αξια =κερδος

και φτιαχνω εναν προσωρινο πινακα απο τον οποι διαλεγω το μεγιστο κερδος το οποι θα μας δειχνει και τον πωλητη που το εφερε αυτο το κερδος

Για να δειξω το ονομα και το αφμ συνδεω τρεις πινακες με inner join τους πινακες sells,sellers,company αυτο το κανω γιατι στον πινακα πωλησεις ειναι το κερδος αν πωλητη και μεσα απο τον πινακα sellers μπορω να παω στον company.

Αυτος ειναι ο πωλητης με το μεγιστο κερδος
"10674" "S091""95168474" "Aundrea Didsbury"

(d) Ποιες είναι οι επισκευές που βρίσκονται σε εκκρεμότητα.

```

select
}
} select id_car,repairs.date_repair_starts
from repairs
where repairs.date_repair_starts not in (select date_ends.date_repair_starts from date_ends )

```

The screenshot shows the pgAdmin 4 interface. The top bar has 'Tools' and 'Help'. Below it is a toolbar with various icons. The main area is a query editor window titled 'postgres on postgres@PostgreSQL_ErgasiaBaseisDedomenwn2018'. It contains the following SQL code:

```

1 select id_car,repairs.date_repair_starts
2 from repairs
3 where repairs.date_repair_starts not in (select date_ends.date_repair_starts from date_ends )

```

Below the code, the status bar shows 'No limit' and other icons. At the bottom of the window is a 'Data Output' tab, which displays the results of the query as a table:

	id_car character (6)	date_repair_starts date
1	C0573	2018-06-04
2	C0932	2018-06-04
3	C0681	2018-06-28
4	C0180	2018-06-23
5	C0878	2018-05-27
6	C0930	2018-06-12
7	C0434	2018-06-11
8	C0989	2018-05-13
9	C0412	2018-06-29
10	C0285	2018-06-03
11	C0594	2018-06-29
12	C0792	2018-06-09
13	C0148	2018-06-15
14	C0680	2018-06-06
15	C0303	2018-05-14
16	C0845	2018-06-10
17	C0094	2018-06-10

```

id_car,repairs.date_repair_starts
from repairs
where repairs.date_repair_starts not in (select date_ends.date_repair_starts from date_ends )

```

Διαλεγω δυο πινακες των πινακα repair και τον πινακα date_ends και συγκρινω τις ημερομηνιες που ξεκιναει η επισκευη με τις ημερομηνιες που τελειωνει αν δεν υπαρχουν τοτε εμφνιζω τα αυτοκινητα και την ημερομηνια που ξεκιναει η επισκευη.

id_car	date_repair_starts
"C0573 "	"2018-06-04"
"C0932 "	"2018-06-04"
"C0681 "	"2018-06-28"
"C0180 "	"2018-06-23"
"C0878 "	"2018-05-27"

"C0930 "	"2018-06-12"
"C0434 "	"2018-06-11"
"C0989 "	"2018-05-13"
"C0412 "	"2018-06-29"
"C0285 "	"2018-06-03"
"C0594 "	"2018-06-29"
"C0792 "	"2018-06-09"
"C0148 "	"2018-06-15"
"C0680 "	"2018-06-06"
"C0303 "	"2018-05-14"
"C0845 "	"2018-06-10"
"C0094 "	"2018-06-10"
"C0817 "	"2018-05-26"
"C0447 "	"2018-05-22"
"C0950 "	"2018-06-23"
"C0124 "	"2018-06-05"
"C0042 "	"2018-06-24"

Είνα 22 αυτά τα αυτοκινητα και διπλα οι ημερομηνιες.

(e) Ποιες είναι οι εργασίες του τεχνικού 'X' τον τελευταία μήνα.

The screenshot shows the pgAdmin 4 interface. At the top, there's a toolbar with various icons like Dashboard, Properties, SQL, Statistics, Dependencies, Dependents, Query - postgres..., Edit Data - Postgres..., and Edit. Below the toolbar is a menu bar with File, Edit, View, Tools, Help, and a status bar at the bottom.

The main area has a title bar "postgres on postgres@PostgreSQL_ErgasiaBaseisDedomenwn2018".

The SQL tab contains the following query:

```

1 select technicians.id_tech ,repairs.date_repair_starts,repairs.id_car,repairs.cost_repair,technicians.afm,company.fullname
2 from repairs
3 inner join technicians
4 on repairs.id_tech=technicians.id_tech
5 inner join company
6 on technicians.afm=company.afm
7 where repairs.date_repair_starts>='2018-6-1' and repairs.date_repair_starts<='2018-6-30'
8 order by repairs.id_tech
9

```

The Data Output tab is selected, displaying the results of the query in a table:

	id_tech character (4)	date_repair_starts date	id_car character (6)	cost_repair integer	afm character (8)	fullname character varying (25)
1	T006	2018-06-03	C0285	1381	16083821	Natassia Ferns
2	T008	2018-06-23	C0180	1300	43763478	Reba Shakle
3	T010	2018-06-04	C0573	527	24027257	Otto Millimoe
4	T015	2018-06-06	C0680	782	38400224	Neely Eberts
5	T017	2018-06-25	C0018	1042	71448119	Charmaine Wheatley
6	T021	2018-06-09	C0792	224	32570356	Alden Blakeborough
7	T023	2018-06-16	C0983	1252	13297823	Emelina Marder
8	T033	2018-06-28	C0681	1517	50771333	Elnora Piper
9	T033	2018-06-24	C0042	918	50771333	Elnora Piper
10	T051	2018-06-23	C0950	1908	15595440	Jenni Stewartson
11	T056	2018-06-15	C0148	1524	2325769	Melissa Proswell
12	T072	2018-06-05	C0124	1001	75851284	Meggi Aldridge
13	T084	2018-06-11	C0434	402	24642862	Almeta Shemwell
14	T084	2018-06-04	C0932	1980	24642862	Almeta Shemwell
15	T086	2018-06-14	C0346	1336	30449501	Rowe Gilogly
16	T086	2018-06-29	C0412	484	30449501	Rowe Gilogly
17	T090	2018-06-29	C0594	1835	53676279	Kennan Allday

```

select
technicians.id_tech ,repairs.date_repair_starts,repairs.id_car,repairs.cost_repair,technicians.afm,compan
y.fullname
from repairs
inner join technicians
on repairs.id_tech=technicians.id_tech
inner join company
on technicians.afm=company.afm

```

```
where repairs.date_repair_starts>='2018-6-1' and repairs.date_repair_starts<='2018-6-30'
order by repairs.id_tech
```

Διαλεγω του τεχνικους απο τον πινακα **technicians** την ημερομηνια που ξεκιναι η επισκευη ,το κοστος επισκευης απο τον πινακα **repairs**,το AFM απο τον πινακα **technicians** και το πληρες ονομα απο τον πινακα **company** σενδεω τους τρεις πινακες **repairs** ,**technicians** ,**company** βλεπω ποιες ειναι οι δουλειες που κανουν οι τεχνικοι απο τον πινακα **repairs** μετα βρισκω αυτους τους τεχνικους απο τον πινακα **technicians** και παιρνω τα ονοματα τους απο τον πινακα **company** τελος βαζω τους περιορισμους να παρω για τον τελευταιο μηνα τις δουλειες >='2018-6-1' εγω εδω εχω διαλεξει υποθετικα οτι ο τελευταιος μηνας ειναι ο Ιουνιος και παιρνω αυτες τις εργασιες και τελος τα κανω order by τα βαζω στην σειρα με βαση τους τεχνικους

Data Output Explain Messages Query History						
	id_tech character (4)	date_repair_starts date	id_car character (6)	cost_repair integer	afm character (8)	fullname character varying (25)
1	T006	2018-06-03	C0285	1381	16083821	Natassia Ferns
2	T008	2018-06-23	C0180	1300	43763478	Reba Shakle
3	T010	2018-06-04	C0573	527	24027257	Otto Millimoe
4	T015	2018-06-06	C0680	782	38400224	Neely Eberts
5	T017	2018-06-25	C0018	1042	71448119	Charmaine Wheatley
6	T021	2018-06-09	C0792	224	32570356	Alden Blakeborough
7	T023	2018-06-16	C0983	1252	13297823	Emelina Marder
8	T033	2018-06-28	C0681	1517	50771333	Elnora Piper
9	T033	2018-06-24	C0042	918	50771333	Elnora Piper
10	T051	2018-06-23	C0950	1908	15595440	Jenni Stewartson
11	T056	2018-06-15	C0148	1524	2325769	Melissa Proswell
12	T072	2018-06-05	C0124	1001	75851284	Meggi Aldridge
13	T084	2018-06-11	C0434	402	24642862	Almeta Shemwell
14	T084	2018-06-04	C0932	1980	24642862	Almeta Shemwell
15	T086	2018-06-14	C0346	1336	30449501	Rowe Gilogly
16	T086	2018-06-29	C0412	484	30449501	Rowe Gilogly
17	T090	2018-06-29	C0594	1835	53676279	Kennan Allday
18	T091	2018-06-10	C0845	996	67886405	Jeanette Ridgway
19	T098	2018-06-10	C0094	221	68664453	Harlan Strapp

και τα αποτελεσματα που παιρνω στο τελος

(f) Ποια είναι τα αυτοκίνητα που έχουν έρθει για επισκευή πάνω από 1 φορά τον τελευταίο χρόνο.

The screenshot shows a pgAdmin interface with a query editor and a results table. The query is as follows:

```
018
1 select count(repair_cars18.id_car) ,repair_cars18.id_car,cars.model
2 from (select repairs.id_car ,repairs.date_repair_starts      from repairs where
3 repairs.date_repair_starts>'2018-1-1'and repairs.date_repair_starts<'2018-12-31')as repair_cars18
4 inner join cars
5 on repair_cars18.id_car=cars.id_car
6 group by  repair_cars18.id_car,cars.model
7 having count(repair_cars18.id_car)>1
8
```

The results table shows the following data:

	count	id_car	model
1	2	C0376	Ram Wagon B350
2	2	C0618	900
3	2	C0388	Legend
4	2	C0573	Quesi
5	2	C0340	Veracruz
6	2	C0284	Escalade ESV
7	2	C0180	Alycone SVX
8	2	C0932	Legend
9	2	C0592	Durango
10	3	C0983	A6
11	3	C0770	Sentra
12	2	C0136	Jetta

```
select count(repair_cars18.id_car) ,repair_cars18.id_car,cars.model
from (select repairs.id_car ,repairs.date_repair_starts      from repairs where
repairs.date_repair_starts>'2018-1-1'and repairs.date_repair_starts<'2018-12-31')as repair_cars18
inner join cars
on repair_cars18.id_car=cars.id_car
group by  repair_cars18.id_car,cars.model
having count(repair_cars18.id_car)>1
```

Εδώ κανω select και εμφανιζω τον αριθμο των αυτοκινητων group by με βαση το id_car απο εναν προσωρινο πινακα που φτιαχνω ποιο κατω και τον ονομαζω με το as **repair_cars18**,επισης παιρνω τα id_car απο τον ιδιο προσωρινο πινακα που φτιαχνω ποιο κατω **repair_cars18** και τα μοντελα απο τον πινακα cars.Αυτος ο νεος προσωρινος πινακας **repair_cars18** εχει τον περιορισμο να παιρνει τις

εργασιες του τελευταιου χρονου εδω για το 2018 και μετα με βαση αυτον τον πινακα μετρω ποσα και ποια αυτοκινητα εχουν ερθει πανω απο μια φορα στον συνεργειο και αυτο το κανω με το group by τα κανω count και με τον περισορισμο **having count(repair_cars18.id_car)>1** παιρνω εκεινα που εχουν ερθει πανω απο μια φορα.

The screenshot shows a PostgreSQL query results window. At the top, there are tabs: Data Output (selected), Explain, Messages, and Query History. The main area displays a table with the following data:

	count bigint	id_car character (6)	model character varying (45)
1	2	C0376	Ram Wagon B350
2	2	C0618	900
3	2	C0388	Legend
4	2	C0573	Quest
5	2	C0340	Veracruz
6	2	C0284	Escalade ESV
7	2	C0180	Alcyone SVX
8	2	C0932	Legend
9	2	C0592	Durango
10	3	C0983	A6
11	3	C0770	Sentra
12	2	C0136	Jetta

Εδω ειναι απαντηση τα αυτοκινητα που ηρθαν πανω απο μια φορα στο συνεργειο.

Ερώτημα 3 (20%). Υλοποίηση triggers και cursors

Αφού ολοκληρώσετε το Ερώτημα 1, απαντήστε στα παρακάτω: (a) Σε αυτό το ερώτημα θα δημιουργήστε ένα trigger (καθώς και τη συνάρτηση που θα εκτελείται όταν ενεργοποιείται το trigger), το οποίο θα υλοποιεί την αυτόματη ενημέρωση του ιστορικού των αυτοκινήτων που διατηρείται στην εταιρεία. Όπως προκύπτει από την περιγραφή της ΒΔ από το ερώτημα 1, υπάρχει πίνακας όπου διατηρείται το ιστορικό του τμήματος επισκευών. Επιπλέον, υπάρχει άλλος ένας πίνακας με τις πληροφορίες των αυτοκινήτων (βλ. ερώτημα 1). Αν έρθει ένα αυτοκίνητο στην εταιρεία για κάποια επισκευή (πίνακας επισκευές) και δεν υπάρχει αντίστοιχη εγγραφή για το συγκεκριμένο αυτοκίνητο (πίνακας αυτοκινητα), τότε ενεργοποιείται ο trigger

```

1  create or replace function add_datatorepairs()
2  returns trigger as
3  $BODY$
4  begin
5  if new.id_car=null then
6  insert into repairs values (new.id_tech,new.id_car,new.cost_repair,new.date_repair_starts);
7  insert into cars values (new.id_car,new.model,new.creation_date,new.car_type_used_or_new);
8  end if;
9  return new;
10 end;
$BODY$
language plpgsql;

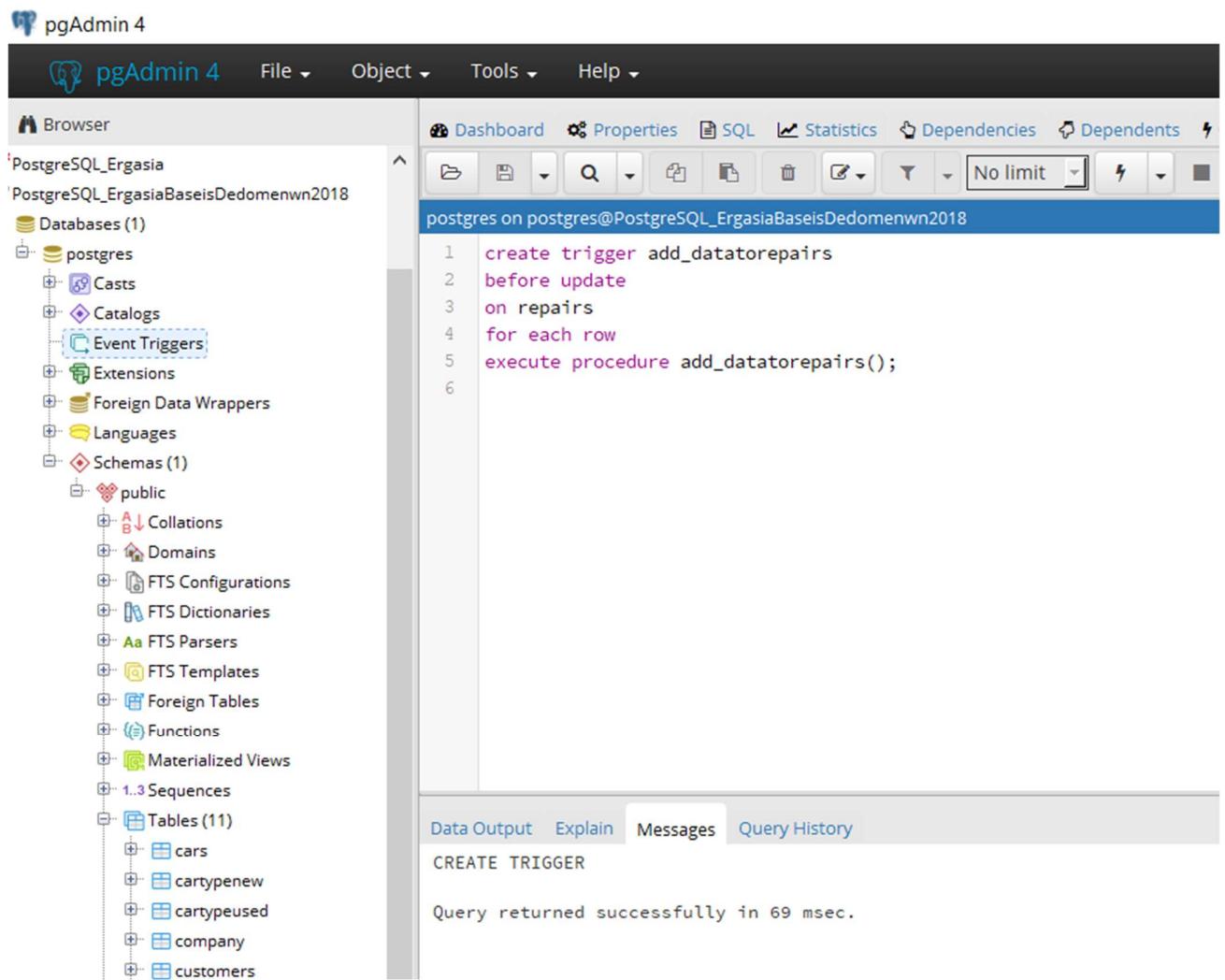
```

The screenshot shows the pgAdmin 4 interface with the following details:

- Left Panel (Browser):** Shows the database structure for 'PostgreSQL_ErgasiaBaseisDedomenwn2018' under the 'postgres' schema, including Databases, Casts, Catalogs, Event Triggers, Extensions, Foreign Data Wrappers, Languages, Schemas (public), Sequences, Tables (11), Trigger Functions (1), Types, and Views.
- Top Bar:** Includes tabs for Dashboard, Properties, SQL, Statistics, Dependencies, Dependents, Query - postgres..., and trigger*.
- Central Area:** Displays the SQL code for creating the trigger function 'add_datatorepairs()'.
- Bottom Panel:** Shows the results of the query execution, indicating a successful CREATE FUNCTION command.

Φτιαχνω μια συναρτηση την οποια ονομαζω add_datatorepairs() αυτη κανει εναν ελεγχο οπου κοιται στον πινακα repairs αν υπαρχει το id_car αν ειναι μηδεν θα πηγαινει και θα κανει insert τα νεα στοιχεια στον πινακα repairs αλλα και στο πινακα cars που εχει ολα τα αυτοκινητα και τις πληροφοριες του

Μετα φτιάχνω ενα trigger το οποιο καλει αυτην την συναρτηση για καθε γραμμη



Ερώτημα 4 (20%). Σύνδεση ΒΔ με JDBC client Υλοποιήστε προγραμματιστικά έναν client σε οποιαδήποτε γλώσσα προγραμματισμού γνωρίζετε (π.χ. Java) χρησιμοποιώντας την κατάλληλη βιβλιοθήκη σύνδεσης με την PostgreSQL (π.χ. JDBC). Ο client θα συνδέεται στο ΣΔΒΔ της PostgreSQL, θα εκτελεί τα queries του Ερωτήματος 2, και θα εμφανίζει τα αποτελέσματα στον χρήστη (είτε σε terminal είτε με γραφικά).

Για να συνδεθω στην βαση δεδομενων θα χρησιμοποιησω την java και το eclipse θα κατεβασω την postgresql connector και ετσι θα συνδεσω το eclipse με την postgresql

<https://jdbc.postgresql.org/download.html>

The screenshot shows a web browser window with multiple tabs open. The active tab is titled "PostgreSQL JDBC Driver". The page content includes:

- A logo for PostgreSQL JDBC Driver.
- A tagline: "The world's most advanced open source database."
- A navigation menu with links: Home, About, Download, Documentation, Community, Development.
- A "Download" section with a sub-menu:
 - About
 - Current Version
 - Supported Versions
 - Other Versions
 - Archived Versions
- An "About" section stating: "Binary JAR file downloads of the JDBC driver are available here and the current version with [Maven Repository](#). Because Java is platform neutral, it is a simple process of just downloading the appropriate JAR file and dropping it into your classpath. Source versions are also available here for recent driver versions."
- A "Current Version 42.2.2" section with a note: "This is the current version of the driver. Unless you have unusual requirements (running old applications or JVMs), this is the driver you should be using. It supports PostgreSQL 8.2 or newer and requires Java 6 or newer. It contains support for SSL and the javax.sql package." Below this are bullet points:
 - If you are using Java 8 or newer then you should use the JDBC 4.2 version.
 - If you are using Java 7 then you should use the JDBC 4.1 version.
 - If you are using Java 6 then you should use the JDBC 4.0 version.
 - If you are using a Java version older than 6 then you will need to use a JDBC3 version of the driver, which will by necessity not be current, found in [Other Versions](#).
- Links to download files:
 - [PostgreSQL JDBC 4.2 Driver, 42.2.2](#) (highlighted with a red oval)
 - [PostgreSQL JDBC 4.1 Driver, 42.2.2.jar](#)
 - [PostgreSQL JDBC 4.0 Driver, 42.2.2.jar](#)
- An "Other Versions" section noting: "Many other versions of the JDBC driver are available. This includes development versions, compatibility with older JDKs, and previous versions of the driver."
- A note about JDK/JVM compatibility: "To determine JDK/JVM compatibility this following list matches up versions of the JVM with the JDBC specification implemented."
- A bullet point: "JDK 1.1 - JDBC 1. Note that with the 8.0 release JDBC 1 support has been removed, so look to update your JDK when you update

Κατεβαζω το αρχειο και το αποθηκευω σε ενα σημειο για να το επιλεξω

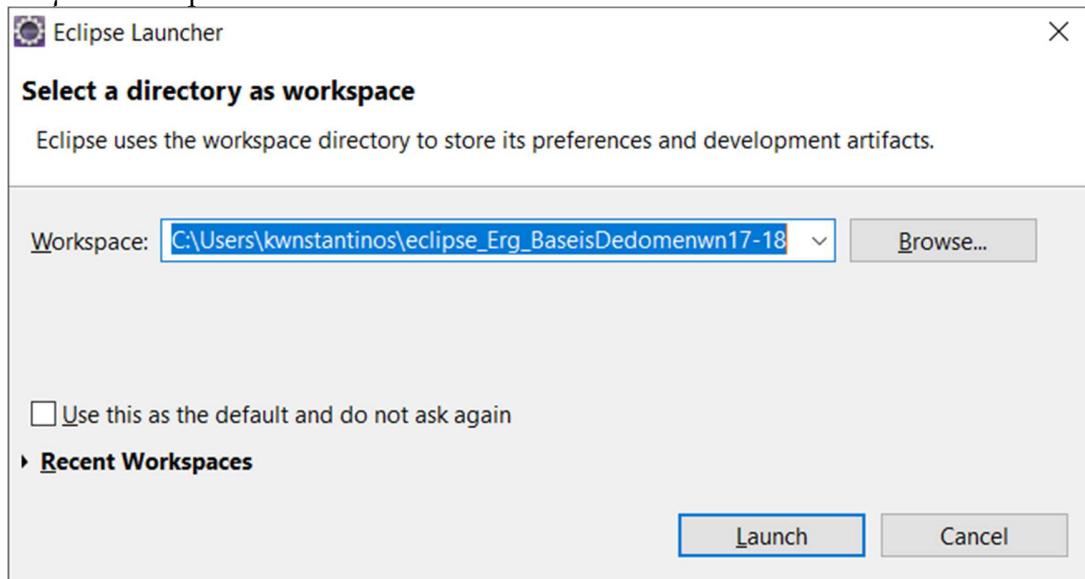
	Name	Date modified	Type	Size
> Quick access	mysql-connector-java-5.1.46	4/28/2018 1:06 PM	File folder	
> OneDrive	annotations-api	4/3/2018 9:04 PM	Executable Jar File	13 KB
▽ This PC	catalina	4/3/2018 9:04 PM	Executable Jar File	1,591 KB
> 3D Objects	catalina-ant	4/3/2018 9:04 PM	Executable Jar File	53 KB
> Desktop	catalina-ha	4/3/2018 9:04 PM	Executable Jar File	116 KB
> Documents	catalina-storeconfig	4/3/2018 9:04 PM	Executable Jar File	75 KB
> Downloads	catalina-tribes	4/3/2018 9:04 PM	Executable Jar File	277 KB
> Music	ecj-4.6.3	4/3/2018 9:04 PM	Executable Jar File	2,393 KB
> Pictures	el-api	4/3/2018 9:04 PM	Executable Jar File	80 KB
> Videos	jasper	4/3/2018 9:04 PM	Executable Jar File	576 KB
> OS (C)	jasper-el	4/3/2018 9:04 PM	Executable Jar File	160 KB
> DATA (D)	jaspic-api	4/3/2018 9:04 PM	Executable Jar File	27 KB
> Network	jsp-api	4/3/2018 9:04 PM	Executable Jar File	61 KB
	mysql-connector-java-5.1.46-bin	2/26/2018 3:28 PM	Executable Jar File	982 KB
	postgresql-42.2.2	5/30/2018 3:22 PM	Executable Jar File	772 KB
	servlet-api	4/3/2018 9:04 PM	Executable Jar File	239 KB
	tomcat-api	4/3/2018 9:04 PM	Executable Jar File	11 KB
	tomcat-coyote	4/3/2018 9:04 PM	Executable Jar File	782 KB
	tomcat-dbcp	4/3/2018 9:04 PM	Executable Jar File	264 KB
	tomcat-i18n-es	4/3/2018 9:04 PM	Executable Jar File	66 KB
	tomcat-i18n-fr	4/3/2018 9:04 PM	Executable Jar File	40 KB
	tomcat-i18n-ja	4/3/2018 9:04 PM	Executable Jar File	42 KB
	tomcat-jdbc	4/3/2018 9:04 PM	Executable Jar File	145 KB
	tomcat-jni	4/3/2018 9:04 PM	Executable Jar File	34 KB
	tomcat-util	4/3/2018 9:04 PM	Executable Jar File	140 KB
	tomcat-util-scan	4/3/2018 9:04 PM	Executable Jar File	204 KB
	tomcat-websocket	4/3/2018 9:04 PM	Executable Jar File	220 KB
	websocket-api	4/3/2018 9:04 PM	Executable Jar File	37 KB

28 items 1 item selected 771 KB



Αποθηκευω το αρχειο και θα το χρησιμοποιησω στο eclipse

Φτιαχνω εναν φακελο με το ονομα C:\Users\kwnstantinos\eclipse_Erg_BaseisDedomenwn17-18 και το ανοιγω στο eclipse.



```

eclipse_Erg_BaseisDedomenwn17-18 - ErgBaseisDedomenwn17-18/src/BaseisDedomenwnErg.java - Eclipse
File Edit Source Refactor Navigate Search Project Run Window Help
Project Explorer BaseisDedomenwnErg.java
ErgBaseisDedomenwn17-18 JRE System Library [JavaSE-1.8]
src
  (default package)
    BaseisDedomenwnErg.java
Referenced Libraries
  postgresql-42.2.2.jar
  D:\myWebProject\apache-tomcat-8.5

171
172
173
174
175
176
177
178
179
180
181
182
183
184
185
186
187
188
189
190
191
192
193
194
195
196
197
198
199
200

System.out.println("f)The cars that have come for repairs more than one time this month are
while ( rs.next() ) {
    int count = rs.getInt("count");
    String id_car=rs.getString("id_car");
    String model=rs.getString("model");

    System.out.println( "count time of repairs = " + count );
    System.out.println( "id_car = " + id_car );
    System.out.println( "model = " + model );

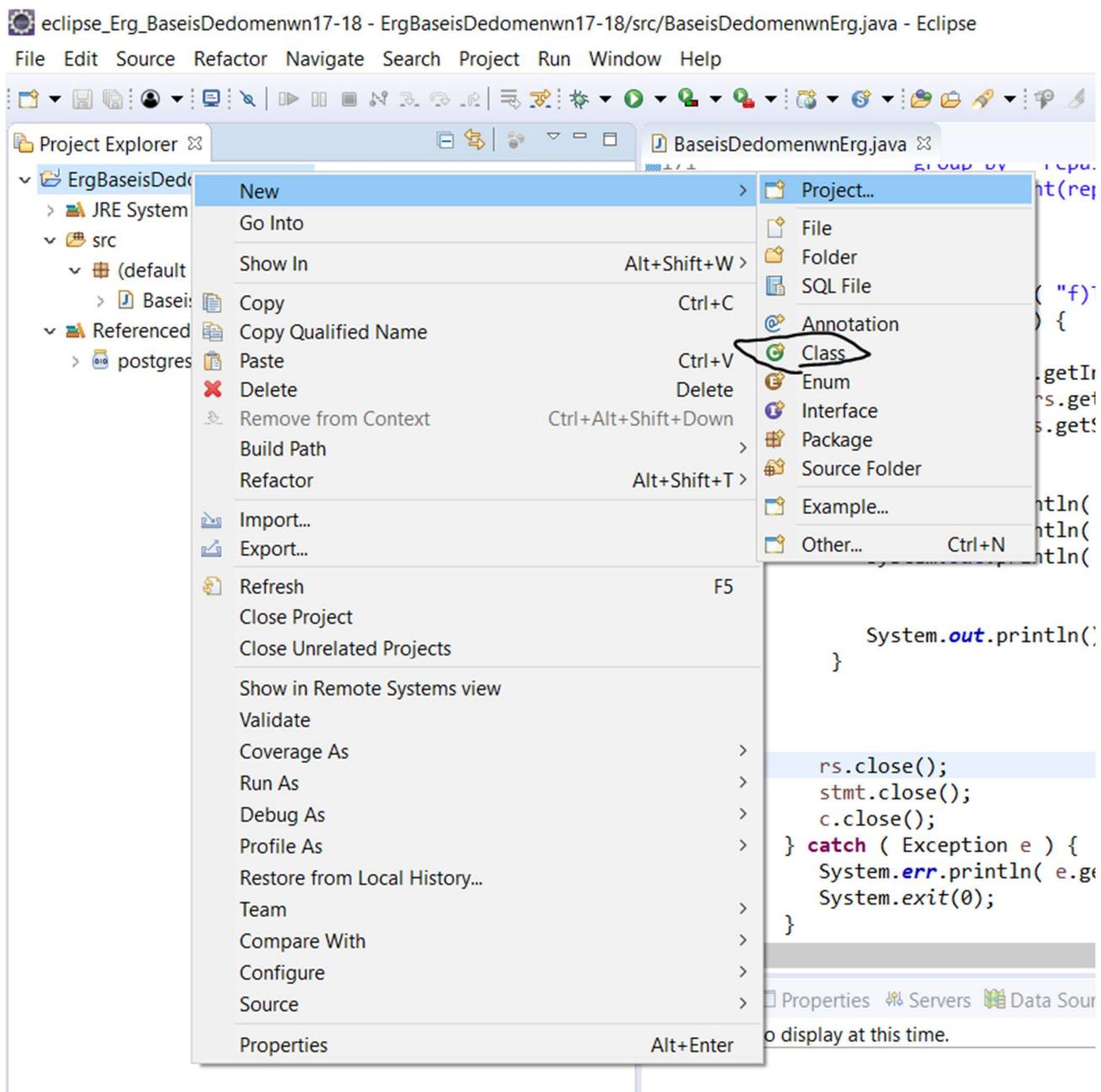
    System.out.println();
}
rs.close();
stmt.close();
c.close();
} catch ( Exception e ) {
    System.err.println( e.getClass().getName()+" "+ e.getMessage() );
    System.exit(0);
}

```

Markers Properties Servers Data Source Explorer Snippets Console

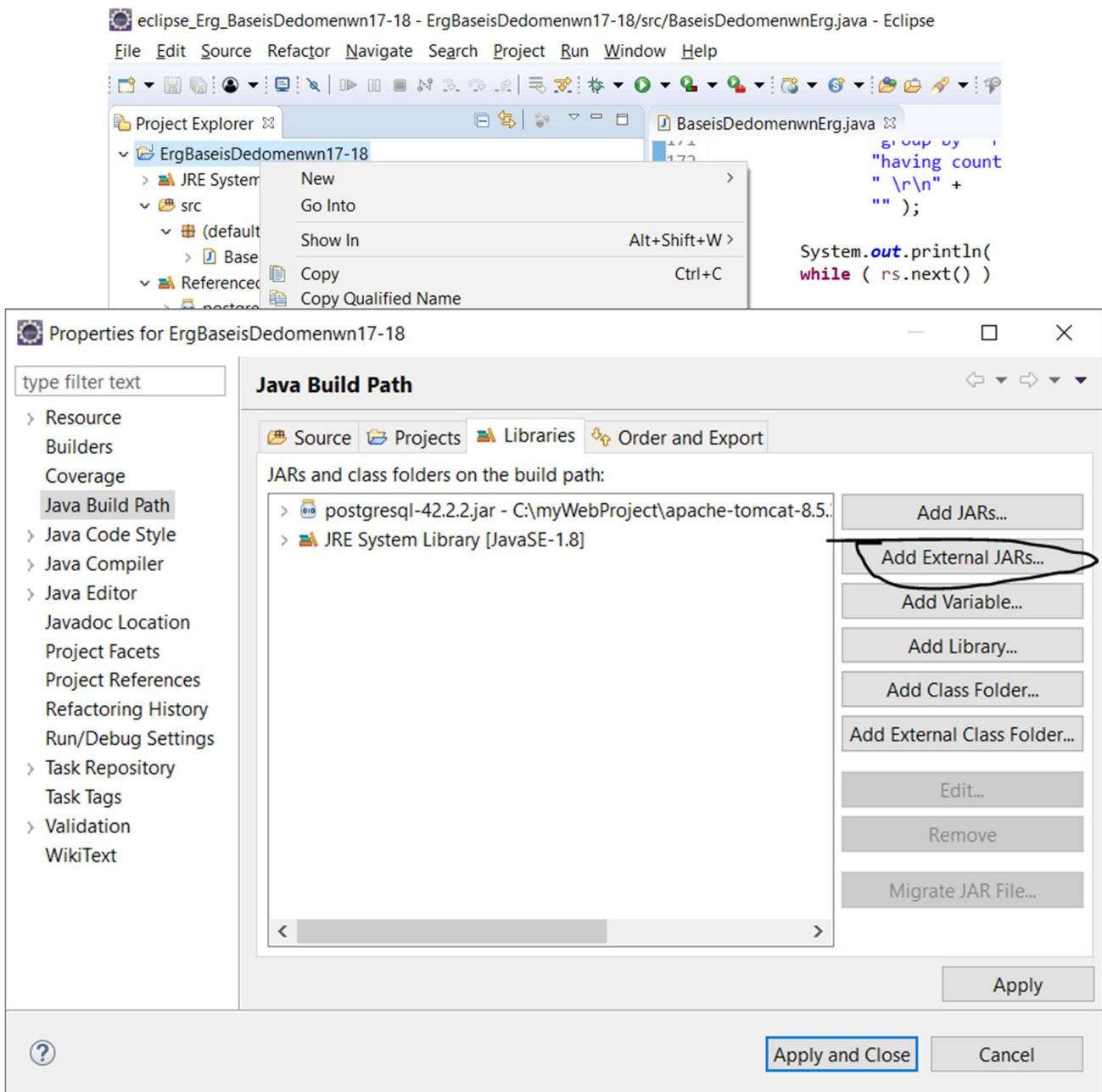
No consoles to display at this time.

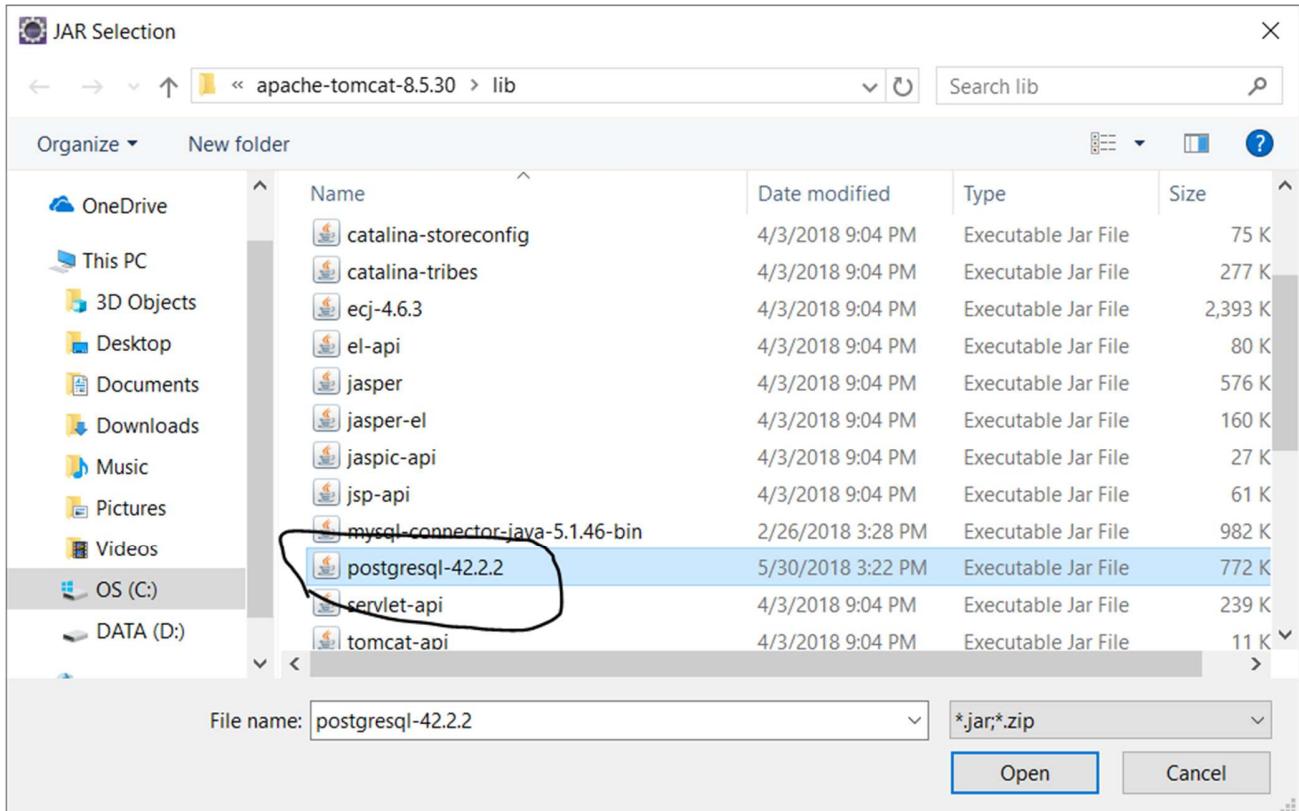
Παταω δεξι click new class



Με ονομα BaseisDedomenwnErg.java

και για να συνδεσω την connector κανω build path





Η πηγή που χρησιμοποιησα είναι

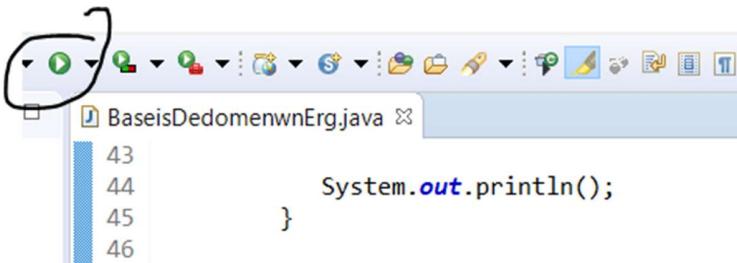
https://www.tutorialspoint.com/postgresql/postgresql_java.htm

```
1④ import java.sql.Connection;⑤
5
6 //p14086kousounnisKwnstantinos
7 //p14036GiwrgosDiwtis
8
9 public class BaseisDedomenwnErg {
10    public static void main( String args[] ) {
11        Connection c = null;
12        Statement stmt = null;
13        try {
14            Class.forName("org.postgresql.Driver");
15            c = DriverManager
16                .getConnection("jdbc:postgresql://localhost:5432/postgres",
17                    "postgres", "pappoulis13");
18            c.setAutoCommit(false);
19            System.out.println("Opened database successfully");
20        } catch (Exception e) {
21            System.out.println(e.getMessage());
22        }
23    }
24}
```

```
c = DriverManager  
    .getConnection("jdbc:postgresql://localhost:5432/postgres",  
    "postgres", "pappoulis13");
```

Συνδεομαι στην βαση δεδομενων μου με username= postgres και password =pappoulis13 εδω σε αυτο το σημειο θα πρεπει για να τρεξει το αρχειο να εχει το credentials τις αντιστοιχης βασης δεδομενων δηλαδη αυτο το username και password πρεπει να τα αλλαξουμε .Εαν δεν υπαρχει η postgres βαση δεδομενων θα φτιαχτει με την πρωτη φορα που θα το τρεξουμε

Τρέχω το πρώτο query



```

43         System.out.println();
44     }
45
46

System.out.println("Fitsr Query answer (a)");
stmt = c.createStatement();
ResultSet rs = stmt.executeQuery( "select repairs.id_car,sum(cost_repair),cars.model\r\n" +
    "from repairs \r\n" +
    "inner join cars\r\n" +
    "on repairs.id_car=cars.id_car\r\n" +
    "group by repairs.id_car,cars.model\r\n" +
    "having sum(cost_repair) in(\r\n" +
    "select max(sum_cost)\r\n" +
    "from(select id_car,sum(cost_repair)as sum_cost from repairs group by id_car)as template)" );

System.out.println( "a)The cars with the maximum cost repair are: \n" );
while ( rs.next() ) {

    String id_car = rs.getString("id_car");
    int sum_cost_repair=rs.getInt("sum");
    String model=rs.getString("model");

    System.out.println( "id_car = " + id_car );
    System.out.println( "sum(cost_repair) = " + sum_cost_repair );
    System.out.println( "Model = " + model );
}

```

Παίρνω τα αποτελεσμάτα με την επαναληψη

```

while(rs.next){
    και τα βαζώ μεσα σε String τα οποια τα τυπωνω
}

```

Κανω την ιδια διαδικασια για ολα

Βλεπουμε τα αποτελεσματα στην κονσολα

```

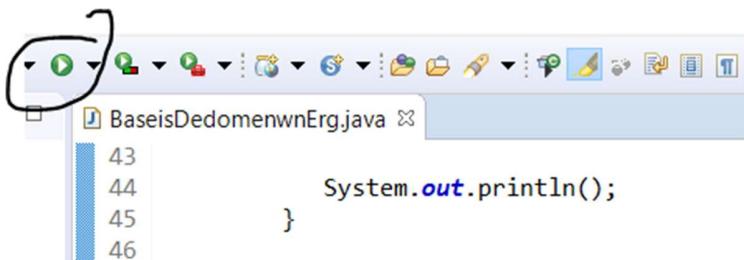
<terminated> BaseisDedomenwnErg [Java Application] C:\Program Files\Java\jre1.
Opened database successfully
Fitsr Query answer (a)
a)The cars with the maximum cost repair are:

id_car = C0136
sum(cost_repair) = 12621
Model = Jetta

id_car = C0472
sum(cost_repair) = 12621
Model = Suburban 2500

```

To τρέχω πανω με το play



```
System.out.println("Second Query answer (b)");
stmt = c.createStatement();
rs = stmt.executeQuery( " select date_trunc('month',date_repair_starts),avg(cost_repair)as Avg_earnings_bymonth\r\n" +
    "from repairs\r\n" +
    "group by date_trunc('month', date_repair_starts)\r\n" +
    "order by date_trunc('month', date_repair_starts)" );

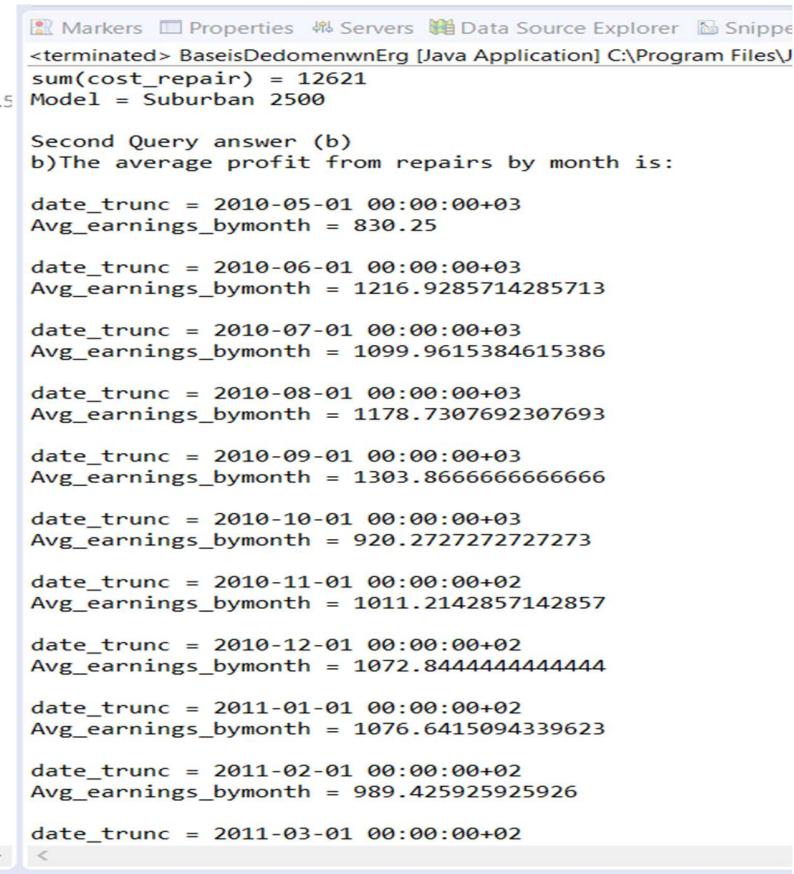
System.out.println( "b)The average profit from repairs by month is: \n" );
while ( rs.next() ) {

    String date_trunc = rs.getString("date_trunc");
    double Avg_earnings_bymonth=rs.getDouble("Avg_earnings_bymonth");

    System.out.println( "date_trunc = " + date_trunc );
    System.out.println( "Avg_earnings_bymonth = " + Avg_earnings_bymonth );

    System.out.println();
}
```

Τα ποτελεσματα στην κονσολα



The screenshot shows a Java application window titled "BaseisDedomenwnErg [Java Application]". The console output displays the following SQL query results:

```
<terminated> BaseisDedomenwnErg [Java Application] C:\Program Files\J
sum(cost_repair) = 12621
Model = Suburban 2500
at-8.5

Second Query answer (b)
b)The average profit from repairs by month is:

date_trunc = 2010-05-01 00:00:00+03
Avg_earnings_bymonth = 830.25

date_trunc = 2010-06-01 00:00:00+03
Avg_earnings_bymonth = 1216.9285714285713

date_trunc = 2010-07-01 00:00:00+03
Avg_earnings_bymonth = 1099.9615384615386

date_trunc = 2010-08-01 00:00:00+03
Avg_earnings_bymonth = 1178.7307692307693

date_trunc = 2010-09-01 00:00:00+03
Avg_earnings_bymonth = 1303.8666666666666

date_trunc = 2010-10-01 00:00:00+03
Avg_earnings_bymonth = 920.2727272727273

date_trunc = 2010-11-01 00:00:00+02
Avg_earnings_bymonth = 1011.2142857142857

date_trunc = 2010-12-01 00:00:00+02
Avg_earnings_bymonth = 1072.8444444444444

date_trunc = 2011-01-01 00:00:00+02
Avg_earnings_bymonth = 1076.6415094339623

date_trunc = 2011-02-01 00:00:00+02
Avg_earnings_bymonth = 989.425925925926

date_trunc = 2011-03-01 00:00:00+02
```

Δεν εχω βαλει ολες τις απαντησεις εχω βαλει στο screenshot ενδεικτικα καποιες εχει τιμες και ποιο κατω.

Τρεχω το τριτο

```

System.out.println("Third Query answer (c)");
stmt = c.createStatement();
rs = stmt.executeQuery( "select sum(car_sold-car_value) ,sells.id_seller,sellers.afm,company.Fullname\r\n" +
"from sells \r\n" +
"inner join sellers\r\n" +
"on sells.id_seller=sellers.id_seller\r\n" +
"inner join company\r\n" +
"on sellers.afm=company.afm\r\n" +
"group by sells.id_seller,sellers.afm,company.Fullname\r\n" +
"having sum(car_sold-car_value)in(\r\n" +
"select max(difference)\r\n" +
"from(select id_seller,sum(car_sold-car_value)as difference from sells group by id_seller)as template)\r\n" +
"" );
System.out.println( "c)Who is the seller with the biggest profits: \n" );
while ( rs.next() ) {
    int Sum = rs.getInt("Sum");
    String id_seller=rs.getString("id_seller");
    double afm=rs.getDouble("afm");
    String fullname=rs.getString("fullname");

    System.out.println( "Sum profits = " + Sum );
    System.out.println( "id_seller = " + id_seller );
    System.out.println( "afm = " + afm );
    System.out.println( "fullname = " + fullname );

    System.out.println();
}

```

The screenshot shows an IDE interface with a code editor and a terminal window. The code editor contains Java code for executing SQL queries and printing results. The terminal window displays the execution of the code, showing the output of the third query (c) and the fourth query (d). The output for query (c) shows the seller with the biggest profits: Aundrea Didsbury, with sum profits 10674, id_seller S091, afm 9.5168474E7, and fullname Aundrea Didsbury.

```

Markers Properties Servers Data Source Explorer Snippet
<terminated> BaseisDedomenwnErg [Java Application] C:\Program Files\Ja
Third Query answer (c)
c)Who is the seller with the biggest profits:

Sum profits = 10674
id_seller = S091
afm = 9.5168474E7
fullname = Aundrea Didsbury

Fourth Query answer (d)

```

```

System.out.println("Fourth Query answer (d)");
stmt = c.createStatement();
rs = stmt.executeQuery( " select id_car,repairs.date_repair_starts \r\n" +
    "from repairs\r\n" +
    "where  repairs.date_repair_starts not in (select date_ends.date_repair_starts from date_ends ) " );

System.out.println( "d)The repairs which are not completed yet: \n" );
while ( rs.next() ) {

    String id_car = rs.getString("id_car");
    String date_repair_starts=rs.getString("date_repair_starts");

    System.out.println( "id_car = " + id_car );
    System.out.println( "date_repair_starts = " + date_repair_starts );

    System.out.println();
}

        Fourth Query answer (d)
d)The repairs which are not completed yet:
id_car = C0573
date_repair_starts = 2018-06-04

id_car = C0932
date_repair_starts = 2018-06-04

id_car = C0681
date_repair_starts = 2018-06-28

id_car = C0180
date_repair_starts = 2018-06-23

id_car = C0878
date_repair_starts = 2018-05-27

id_car = C0930
date_repair_starts = 2018-06-12

id_car = C0434
date_repair_starts = 2018-06-11

id_car = C0989
date_repair_starts = 2018-05-13

id_car = C0412
date_repair_starts = 2018-06-29

id_car = C0285
date_repair_starts = 2018-06-03

id_car = C0594
<

```

Μερικες απο τις τιμες στην κονσολα δεν τις δειχνω ολες πρεπει να κανω scroll στην κονσολα

```

System.out.println("Fifth Query answer (e)");
stmt = c.createStatement();
rs = stmt.executeQuery( "select technicians.id_tech ,repairs.date_repair_starts,repairs.id_car,repairs.cost_repair,technicians.afm,company.fullname\r\n" +
"from repairs\r\n" +
"inner join technicians\r\n" +
"on repairs.id_tech=technicians.id_tech\r\n" +
"inner join company\r\n" +
"on technicians.afm=company.afm\r\n" +
"where repairs.date_repair_starts>='2018-6-1' and repairs.date_repair_starts<='2018-6-30'\r\n" +
"order by repairs.id_tech\r\n" +
" " );
System.out.println( "e)The jobs of 'x' technician the last month : \n" );
while ( rs.next() ) {

    String id_tech = rs.getString("id_tech");
    String date_repair_starts = rs.getString("date_repair_starts");
    String id_car = rs.getString("id_car");
    int cost_repair = rs.getInt("cost_repair");
    double afm = rs.getDouble("afm");
    String fullname = rs.getString("fullname");

    System.out.println( "id_car = " + id_tech );
    System.out.println( "date_repair_starts = " + date_repair_starts );
    System.out.println( "id_car = " + id_car );
    System.out.println( "cost_repair = " + cost_repair );
    System.out.println( "afm = " + afm );
    System.out.println( "fullname = " + fullname );

    System.out.println();
}

```

```

127
-8.5 Markers Properties Servers Data Source Explorer Sni
<terminated> BaseisDedomenwnErg [Java Application] C:\Program Fil
date_repair_starts = 2018-06-24

Fifth Query answer (e)
e)The jobs of 'x' technician the last month :

id_car = T006
date_repair_starts = 2018-06-03
id_car = C0285
cost_repair = 1381
afm = 1.6083821E7
fullname = Natassia Ferns

id_car = T008
date_repair_starts = 2018-06-23
id_car = C0180
cost_repair = 1300
afm = 4.3763478E7
fullname = Reba Shakle

id_car = T010
date_repair_starts = 2018-06-04
id_car = C0573
cost_repair = 527
afm = 2.4027257E7
fullname = Otto Millimoe

id_car = T015
date_repair_starts = 2018-06-06
id_car = C0680
cost_repair = 782
afm = 3.8400224E7
fullname = Neely Eberts

```

Κανω screenshot καποιες αρ τιμες ενδεικτικα στην κονσολα βγαινουν όλες

Τρεχω το 60

```

        System.out.println();
    }

    System.out.println("Sixth Query answer (f)");
    stmt = c.createStatement();
    rs = stmt.executeQuery( " select count(repair_cars18.id_car) ,repair_cars18.id_car,cars.model\r\n" +
    "from (select repairs.id_car ,repairs.date_repair_starts   from repairs where repairs.date_repair_starts>'2018-1-1'and repairs.date_repair_starts<'2018-12-31')as repair_cars18\r\n" +
    "inner join cars\r\n" +
    "on repair_cars18.id_car=cars.id_car\r\n" +
    "group by  repair_cars18.id_car,cars.model\r\n" +
    "having count(repair_cars18.id_car)>1\r\n" +
    " \r\n" +
    " " );
}

System.out.println( "f)The cars that have come for repairs more than one time this month are: \n" );
while ( rs.next() ) {

    int count = rs.getInt("count");
    String id_car=rs.getString("id_car");
    String model=rs.getString("model");

    System.out.println( "count time of repairs = " + count );
    System.out.println( "id_car = " + id_car );
    System.out.println( "model = " + model );

    System.out.println();
}

rs.close();
stmt.close();
c.close();
} catch ( Exception e ) {
    System.err.println( e.getClass().getName()+" "+ e.getMessage() );
    System.exit(0);
}
System.out.println("Operation done successfully");

```

Updates Available
Updates are available for your soft

Κανω screenshot καποιες αο τις τιμες ενδεικτικα στην κουσολα βγαινουν όλες

The screenshot shows an IDE interface with the following details:

- Title Bar:** Shows the file number "165" and the code line "System.out.println("Sixth Query answer (f)");".
- Toolbar:** Includes "Markers", "Properties", "Servers", "Data Source Explorer", "Snippets", and "Console".
- Status Bar:** Displays "terminated> BaseisDedomenwnErg [Java Application] C:\Program Files\Java\jre1.8.0_161\bin\javaw.exe (May 2020)".
- Output Window:** Shows the execution results of the Java code. It lists multiple cars that have been repaired more than once in the current month, with their ID, model, and repair count. The results are as follows:

Repair Count	ID Car	Model
2	C0376	Ram Wagon B350
2	C0618	900
2	C0388	Legend
2	C0573	Quest
2	C0340	Veracruz
2	C0284	Escalade ESV
2	C0180	Alcyone SVX
2	<	