

**Ανοικτά Ακαδημαϊκά Μαθήματα**

**Τεχνολογικό Εκπαιδευτικό Ίδρυμα Αθήνας**

Βάσεις Δεδομένων ΙΙ (Ε)

**Ενότητα 1:** Επισκόπηση θεμάτων σχεδιασμού Βάσεων Δεδομένων και υλοποίησης με χρήση SQL

Χ. Σκουρλάς, Α. Τσολακίδης, Δ. Βάσσης

Τμήμα Μηχανικών Πληροφορικής ΤΕ

|  |  |
| --- | --- |
| Το περιεχόμενο του μαθήματος διατίθεται με άδεια Creative Commons εκτός και αν αναφέρεται διαφορετικά | Το έργο υλοποιείται στο πλαίσιο του Επιχειρησιακού Προγράμματος «Εκπαίδευση και Δια Βίου Μάθηση» και συγχρηματοδοτείται από την Ευρωπαϊκή Ένωση (Ευρωπαϊκό Κοινωνικό Ταμείο) και από εθνικούς πόρους. |

Περιεχόμενα

[1ο. ΘΕΜΑ (σχεδιασμός βάσης δεδομένων) 3](#_Toc401491576)

[2ο. ΘΕΜΑ (υλοποίηση βάσης δεδομένων)) 5](#_Toc401491577)

Στόχος του πρώτου εργαστηρίου είναι μία περιεκτική εστίαση σε σημαντικά θέματα σχεδιασμού Βάσεων Δεδομένων (μοντελοποίηση, κανονικοποίηση) και υλοποίησης Βάσεων Δεδομένων με χρήση SQL.

**Λέξεις Κλειδιά**: Μοντελοποίηση, Κανονικοποίηση, Μοντέλο Οντοτήτων Συσχετίσεων, SQL, Κανόνες ακεραιότητας

Τα θέματα αυτά αποτέλεσαν αντικείμενο μελέτης στο πλαίσιο του μαθημάτων «Βάσεων Δεδομένων Ι». Το υλικό βασίζεται σε επεξεργασία των θεμάτων που τέθηκαν στις Εξετάσεις Χειμερινού εξαμήνου στο μάθημα «Βάσεις Δεδομένων Ι – Θεωρία» (Υπεύθυνος καθηγητής: Χ. Σκουρλάς, ημερομηνία εξέτασης: 17/2/2014).

# ΘΕΜΑ (σχεδιασμός βάσης δεδομένων)

Η βάση δεδομένων των υπαλλήλων της εταιρείας *e-*ΚΟΣΜΟΣ περιλαμβάνει τα παρακάτω στοιχεία:

Empno=Κωδικός υπαλλήλου, Name=όνομα, JobNo=κωδικός θέσης, Job=θέση, Deptno=κωδικός τμήματος, Dname=τμήμα Sal=μισθός, C\_No=αριθμός παιδιών υπαλλήλου, C\_Name=όνομα παιδιού, B\_Date= ημερομηνία γέννησης παιδιού.

**Περιορισμοί**

Υποτίθεται ότι κάθε υπάλληλος έχει μία θέση, ανήκει σε ένα τμήμα, ο μισθός του εξαρτάται από τη θέση και έχει ή δεν έχει παιδιά.

Employee

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Empno | Name | JobNo | Job | DeptNo | Dname | Sal | C\_no | C\_Name | B\_date |
| 10 | ΣΠΥΡΟΥ | 100 | ΠΩΛΗΤΗΣ | 50 | ΠΩΛΗΣΕΙΣ | 2200 | 2 | ΜΑΡΙΑ | 10/01/89 |
| 10 | ΣΠΥΡΟΥ | 100 | ΠΩΛΗΤΗΣ | 50 | ΠΩΛΗΣΕΙΣ | 2200 | 2 | ΙΩΑΝΝΗΣ | 20/03/90 |
| 20 | ΧΡΗΣΤΟΥ | 200 | ΑΝΑΛΥΤΗΣ | 60 | ΛΟΓΙΣΤΗΡΙΟ | 2000 |  |  |  |
| 30 | ΝΙΚΟΥ | 300 | ΧΕΙΡΙΣΤΗΣ | 70 | ΜΙΣΘΟΔΟΣΙΑ | 1000 | 1 | ΘΩΜΑΣ | 10/06/89 |

Γράψτε Πρώτη, Δεύτερη και Τρίτη Κανονική Μορφή και σχεδιάστε το Μοντέλο Οντοτήτων–Συσχετίσεων με συμβολισμό Navathe-Elmasri.

**Υπόδειξη επίλυσης**

Ποιο είναι το κύριο κλειδί στην 1NF; Σίγουρα όχι το σύνθετο (empno, c\_name) (γιατί;)

**Μοντέλο Οντοτήτων Συσχετίσεων (Entity Relationship model)**



**Τρίτη Κανονική Μορφή 3NF**

Employee

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Empno | Name | JobNo | DeptNo | C\_no |
| 10 | ΣΠΥΡΟΥ | 100 | 50 | 2 |
| 20 | ΧΡΗΣΤΟΥ | 200 | 60 |  |
| 30 | ΝΙΚΟΥ | 300 | 70 | 1 |

*Κύριο κλειδί: empno*

Jobs Dept

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| JobNo | Job | Sal |  |  | DeptNo | Dname |
| 100 | ΠΩΛΗΤΗΣ | 2200 |  |  | 50 | ΠΩΛΗΣΕΙΣ |
| 200 | ΑΝΑΛΥΤΗΣ | 2000 |  |  | 60 | ΛΟΓΙΣΤΗΡΙΟ |
| 300 | ΧΕΙΡΙΣΤΗΣ | 1000 |  |  | 70 | ΜΙΣΘΟΔΟΣΙΑ |

*Κύριο κλειδί: JobNo Κύριο κλειδί: deptno*

Child

|  |  |  |
| --- | --- | --- |
| Empno | C\_Name | B\_date |
| 10 | ΜΑΡΙΑ | 10-JAN-89 |
| 10 | ΙΩΑΝΝΗΣ | 20-MAR-90 |
| 30 | ΘΩΜΑΣ | 10-JUN-89 |

*Κύριο κλειδί: (empno, c\_name)*

Names

|  |
| --- |
| C\_Name |
| ΘΩΜΑΣ |
| ΙΩΑΝΝΗΣ |
| ΜΑΡΙΑ |

*Κύριο κλειδί: (c\_name)*

**Παρατήρηση:** Ο τελευταίος πίνακας θα μπορούσε να λείπει. Πότε μας είναι απαραίτητος;

# ΘΕΜΑ (υλοποίηση βάσης δεδομένων))

**Υπόδειξη επίλυσης**

Έστω απλοποιημένη βάση δεδομένων Διεύθυνσης Προσωπικού. Οι στήλες των πινάκων είναι: Empno=Κωδικός υπαλλήλου, Name=όνομα, JobCode/JobNo=κωδικός θέσης, Job\_descr= θέση, Deptno=κωδικός τμήματος, Dname=τμήμα Sal=μισθός, Comm=προμήθεια. Υποτίθεται ότι κάθε υπάλληλος ανήκει σε ένα τμήμα και ότι ο μισθός του εξαρτάται από τη θέση του.

 Emp

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Empno | Name | JobΝο | DeptNo | Comm |
| 10 | ΣΠΥΡΟΥ | 100 | 50 | 450 |
| 20 | ΧΡΗΣΤΟΥ | 200 | 50 |  |
| 30 | ΝΙΚΟΥ | 300 | 60 |  |

 Job

|  |  |  |
| --- | --- | --- |
| JobCode | Job\_descr | Sal |
| 100 | ΠΩΛΗΤΗΣ | 2200 |
| 200 | ΑΝΑΛΥΤΗΣ | 2000 |
| 300 | ΧΕΙΡΙΣΤΗΣ | 1000 |

Dept

|  |  |  |
| --- | --- | --- |
| DeptNo | Dname | Loc |
| 50 | ΠΩΛΗΣΕΙΣ | ΑΘΗΝΑ |
| 60 | ΛΟΓΙΣΤΗΡΙΟ | ΑΘΗΝΑ |
| 70 | ΜΙΣΘΟΔΟΣΙΑ | ΒΟΛΟΣ |

**Μετά από κάθε ερώτηση ακολουθούν απαντήσεις σπουδαστών. Σχολιάστε τις απαντήσεις αυτές.**

1. Δημιουργήστε τη βάση με κύρια και ξένα κλειδιά

/\* Πρώτη εκδοχή \*/

CREATE DATABASE THEMA;

USE THEMA;

CREATE TABLE DEPT(DEPTNO INT(2) NOT NULL, DNAME VARCHAR(30),

 LOC VARCHAR(30),

PRIMARY KEY(DEPTNO));

CREATE TABLE JOB(JOBCODE INT(3) NOT NULL, JOB\_DESCR VARCHAR(30),

 SAL INT(4),

PRIMARY KEY (JOBCODE));

CREATE TABLE EMP(EMPNO INT(2) NOT NULL, ENAME VARCHAR(30),

 JOBNO INT(3), DEPTNO INT(2), COMM INT(4),

PRIMARY KEY (EMPNO),

FOREIGN KEY (JOBNO) REFERENCES JOB(JOBCODE),

FOREIGN KEY (DEPTNO) REFERENCES DEPT(DEPTNO));

/\* Δεύτερη εκδοχή \*/

drop database if exists testing;

create database testing;

use testing;

create table Job (JobCode int(4) not null, Job\_descr varchar(25),

 Sal int (6), primary key (Jobcode));

create table Dept (Deptno int(3) not null, Dname varchar(25),

 Loc varchar(25), primary key (Deptno));

create table Emp (Empno int(4) not null, Name varchar(20),

 JobNo int(4), Deptno int(3), Comm int (6),

 primary key (Empno),

 foreign key (JobNo) references Job(JobCode),

 foreign key (DeptNo) references Dept(DeptNo));

1. Γράψτε τις απαραίτητες εντολές INSERT INTO ώστε να καταχωρηθούν όλα τα στοιχεία του ΣΠΥΡΟΥ.

/\* Πρώτη εκδοχή \*/

INSERT INTO DEPT VALUES

(50,'ΠΩΛΗΣΕΙΣ','ATHINA');

INSERT INTO JOB VALUES

(100,'ΠΩΛΗΤΗΣ',2200);

INSERT INTO EMP VALUES

(10,'ΣΠΥΡΟΥ',100,50,450);

/\* Δεύτερη εκδοχή \*/

insert into Job values (100,'PWLHTHS',2200);

insert into Dept values (50,'PWLHSEIS','ATHENS');

insert into Emp values (10,'SPYROU',100,50,450);

/\* Εισαγωγή και των υπολοίπων στηλών των πινάκων \*/

insert into Job values (200,'ANALYTHS',2000), (300,'XEIRISTHS',1000);

insert into Dept values

 (60,'LOGISTHRIO','ATHENS'),(70,'MISTHODOSIA','VOLOS');

insert into Emp values (20,'HRISTOU',200,50,NULL), (30,'NIKOY',300,60,NULL);

select \* from Job;

select \* from Dept;

select \* from Emp;

1. Δείξτε (SELECT) name, empno, job\_descr, sal, deptno, dname των υπαλλήλων που είναι αναλυτές ή πωλητές ή χειριστές. Οι υπάλληλοι θα είναι ταξινομημένοι ανά θέση (job\_descr).

/\* Πρώτη εκδοχή \*/

SELECT ENAME,EMPNO,JOB\_DESCR,SAL,EMP.DEPTNO,DNAME

FROM EMP,DEPT,JOB

WHERE EMP.DEPTNO=DEPT.DEPTNO

AND JOBNO=JOBCODE

AND JOB\_DESCR IN ('ΠΩΛΗΤΗΣ','ΑΝΑΛΥΤΗΣ','ΧΕΙΡΙΣΤΗΣ')

ORDER BY JOB\_DESCR;

/\* Δεύτερη εκδοχή \*/

select empno,job\_descr,sal,emp.deptno,dname

from emp,job,dept

where emp.deptno=dept.deptno

and emp.jobNo=Job.JobCode

and job\_descr in ('PWLHTHS','ANALYTHS','XEIRISTHS')

order by job\_descr;

1. Δείξτε name, empno, job\_descr, sal των υπαλλήλων με σύνολο αμοιβών (άθροισμα μισθού και προμήθειας) μεγαλύτερο ή ίσο των 1000 ευρώ και μικρότερο ή ίσο των 3000 ευρώ (τουλάχιστον με 2 τρόπους).

/\* Πρώτη εκδοχή \*/

SELECT ENAME,EMPNO,JOB\_DESCR,SAL

FROM EMP,JOB

WHERE JOBNO=JOBCODE

AND ( (SAL+IFNULL(COMM,0))>=1000 AND (SAL+IFNULL(COMM,0))<=3000 );

SELECT ENAME,EMPNO,JOB\_DESCR,SAL

FROM EMP,JOB

WHERE JOBNO=JOBCODE

AND SAL+IFNULL(COMM,0) BETWEEN 1000 AND 3000;

/\* Δεύτερη εκδοχή \*/

select name,empno,job\_descr,sal

from emp,job

where Emp.jobNo=Job.jobCode

and (ifnull(emp.comm,0)+sal) >= 1000

and (ifnull(emp.comm,0)+sal) <=3000;

select name,empno,job\_descr,sal

from emp,job

where Emp.jobNo=Job.jobCode

and (ifnull(emp.comm,0)+sal) between 1000 and 4000;

1. Διορθώστε και συμπληρώστε την παρακάτω εντολή SELECT ώστε να δείχνει στοιχεία πωλητών, αναλυτών και χειριστών που έχουν μισθό μεγαλύτερο των 1500 ευρώ και μικρότερο από 4000 ευρώ. Πρέπει να έχουμε δύο επίπεδα ταξινόμησης: Οι υπάλληλοι θα τυπώνονται ανά θέση δηλαδή πρώτα οι αναλυτές, μετά οι πωλητές και μετά οι χειριστές και οι υπάλληλοι που έχουν την ίδια θέση θα εμφανίζονται αλφαβητικά.

SELECT empno, name, job, sal, comm, deptno, dname

FROM

WHERE

ORDER BY

/\* Πρώτη εκδοχή \*/

SELECT EMPNO, ENAME, JOB\_DESCR, SAL, COMM, EMP.DEPTNO, DNAME

FROM EMP,DEPT,JOB

WHERE EMP.DEPTNO=DEPT.DEPTNO

AND JOBNO=JOBCODE

AND JOB\_DESCR IN ('ΠΩΛΗΤΗΣ','ΑΝΑΛΥΤΗΣ','ΧΕΙΡΙΣΤΗΣ')

AND SAL BETWEEN 1500 AND 4000

ORDER BY JOB\_DESCR,ENAME;

/\* Δεύτερη εκδοχή \*/

select empno,name,job\_descr as job,sal,comm,emp.deptno,dname

from emp,job,dept

where emp.deptno=dept.deptno and emp.jobno=job.jobcode

and job\_descr in ('PWLHTHS','ANALYTHS','XEIRISTHS')

and sal>1500 and sal<4000

order by job\_descr,name;

1. Γράψτε εντολή SELECT η οποία υπολογίζει ανά κωδικό θέσης (100, 200, …) πόσοι είναι οι υπάλληλοι που κατέχουν τη θέση. Να πως θα φαίνονται τα αποτελέσματα στο συγκεκριμμένο παράδειγμα.

|  |  |
| --- | --- |
| JobΝο | No\_of\_employees |
| 100 | 1 |
| 200 | 1 |
| 300 | 1 |

/\* Πρώτη εκδοχή \*/

SELECT JOBNO, COUNT(\*) AS NO\_OF\_EMPLOYEES

FROM EMP

GROUP BY JOBNO;

/\* Δεύτερη εκδοχή \*/

select JobNo, count(\*) as No\_of\_employees

from emp

group by jobno;

1. Τροποποιήστε την εντολή SELECT έτσι ώστε αν και υπολογίζει πόσοι είναι οι υπάλληλοι ανά κωδικό θέσης (100, 200, 300, …) να δείχνει μόνο θέσεις που απασχολούν τουλάχιστον έναν υπαλλήλο:

|  |  |
| --- | --- |
| JobNo | No\_of\_employees. |
| 100 | 1 |
| 200 | 1 |
| 300 | 1 |

/\* Πρώτη εκδοχή \*/

SELECT JOBNO, COUNT(\*) AS NO\_OF\_EMPLOYEES

FROM EMP

GROUP BY JOBNO

HAVING COUNT(\*)>=1;

/\* Δεύτερη εκδοχή \*/

select JobNo,count(\*) as No\_of\_employees

from emp

group by jobno

having count(\*)>=1;

1. Εμφάνισε κάθε εργαζόμενο της εταιρείας που έχει μισθό ίσο με τον μεγαλύτερο μισθό στο τμήμα 'ΠΩΛΗΣΕΙΣ'

/\* Πρώτη εκδοχή \*/

SELECT ENAME,EMPNO,SAL

FROM EMP,JOB

WHERE JOBNO=JOBCODE

AND SAL=(SELECT MAX(SAL)

 FROM EMP,DEPT,JOB

 WHERE EMP.DEPTNO=DEPT.DEPTNO

 AND JOBNO=JOBCODE

 AND DNAME='ΠΩΛΗΣΕΙΣ');

/\* Δεύτερη εκδοχή \*/

/\*The join in the nested select is optional I guess..\*/

select empno,name

from emp,dept,job

where emp.deptno=dept.deptno

and emp.jobno=job.jobcode

and sal = (select max(sal) from emp,job,dept where

/\* emp.deptno=dept.deptno and emp.jobno=job.jobcode and \*/ dname='PWLHSEIS');

/\* Τρίτη εκδοχή \*/

SELECT ename, sal, dname

FROM emp,job,dept

WHERE dept.deptno=emp.deptno

AND job.jobcode=emp.jobno

AND sal= ( SELECT MAX(sal)

 FROM job,emp,dept

 WHERE dept.deptno=emp.deptno

 AND job.jobcode=emp.jobno

 AND dname='pwlhseis');

1. Δείξτε όλα τα τμήματα με τριγράμματη συντομογραφία αποτελούμενη από τα τρία πρώτα γράμματα του ονόματός τους

|  |  |  |
| --- | --- | --- |
| DeptNo | Dname | Abbr  |
| 50 | ΠΩΛΗΣΕΙΣ | ΠΩΛ |
| 60 | ΛΟΓΙΣΤΗΡΙΟ | ΛΟΓ |
| 70 | ΜΙΣΘΟΔΟΣΙΑ | ΜΙΣ |

/\* Πρώτη εκδοχή \*/

SELECT DEPTNO,DNAME,SUBSTR(DNAME,1,3) AS ADDR

FROM DEPT;

/\* Δεύτερη εκδοχή \*/

select DeptNo,Dname,substring(Dname,1,3)

from dept;

1. Γράψτε Πρώτο και Δεύτερο Κανόνα ακεραιότητας.

**Σχολιάστε τις παρακάτω απαντήσεις και διατυπώστε με δικό σας τρόπο**

/\* Πρώτη εκδοχή \*/

Πρώτος κανόνας ακεραιότητας:

Καμιά συνιστώσα της τιμής ενός κύριου κλειδιού δεν μπορεί να μην έχει τιμή. Ο κανόνας εξασφαλίζει ότι ανά δύο οι οντότητες (οι εγγραφές) είναι διακεκριμένες.

Δεύτερος κανόνας ακεραιότητας:

Έστω D ένα πρώτο πεδίο ορισμού δηλαδή ένα πεδίο ορισμού επάνω στο

οποίο ορίζεται ένα απλό κύριο κλειδί. Έστω σχέση R1 με ιδιότητα Α που ορίζεται στο D και έστω σχέση R2 με κύριο κλειδί οριζόμενο στο D. Τότε η τιμή της ιδιότητας Α στο R1 πρέπει να είναι είτε τίποτα (null) ή ίση με τιμή του κύριου κλειδιού κάποιας πλειάδας της σχέσης R2 .

/\* Δεύτερη εκδοχή \*/

Entity Integrity Rule:

- The entity integrity rule refers to rules the primary key must follow.

 The primary key value cannot be null.

 The primary key value must be unique.

Referential Integrity Rule:

- The referential integrity rule refers to the foreign key.

 The foreign key may be null and may have the same value but:

 The foreign key value must match a record in the table it is referring to.

/\* Τρίτη εκδοχή \*/

Πρώτος κανόνας ακεραιότητας: Το κύριο κλειδί ή κάποιο μέρος του (αν έχουμε σύνθετο κύριο κλειδί) απαγορεύεται να είναι NULL.

Δεύτερος κανόνας ακεραιότητας: Απαγορεύεται να γίνει INSERT μια τιμή σε μία στήλη που είναι ξένο κλειδί αν αυτή η τιμή δεν υπάρχει εκεί που η αντίστοιχη στήλη της είναι κύριο κλειδί. Αντίστοιχα ισχύουν για UPDATE, DELETE. Στο σχήμα ο πίνακας Alpha έχει μόνο μία γραμμή.

Table Alpha Table B

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| A (Primary key) | columns |   | B(Primary key) | C(Foreign key) | columns |
| 50 | … |  | … | 40 | … |

|  |
| --- |
| **Ανοικτά Ακαδημαϊκά Μαθήματα****Τεχνολογικό Εκπαιδευτικό Ίδρυμα Αθήνας** |
| **Τέλος Ενότητας** |
| **Χρηματοδότηση*** Το παρόν εκπαιδευτικό υλικό έχει αναπτυχθεί στα πλαίσια του εκπαιδευτικού έργου του διδάσκοντα.
* Το έργο «**Ανοικτά Ακαδημαϊκά Μαθήματα στο ΤΕΙ Αθήνας**» έχει χρηματοδοτήσει μόνο τη αναδιαμόρφωση του εκπαιδευτικού υλικού.
* Το έργο υλοποιείται στο πλαίσιο του Επιχειρησιακού Προγράμματος «Εκπαίδευση και Δια Βίου Μάθηση» και συγχρηματοδοτείται από την Ευρωπαϊκή Ένωση (Ευρωπαϊκό Κοινωνικό Ταμείο) και από εθνικούς πόρους.

 |

**Σημειώματα**

**Σημείωμα Αναφοράς**

Copyright ΤΕΙ Αθήνας, Χ. Σκουρλάς, Α. Τσολακίδης, Δ. Βάσσης 2014. Χ. Σκουρλάς, Α. Τσολακίδης, Δ. Βάσσης. «Βάσεις Δεδομένων ΙΙ (Ε). Ενότητα 1: Επισκόπηση θεμάτων σχεδιασμού Βάσεων Δεδομένων και υλοποίησης με χρήση SQL». Έκδοση: 1.0. Αθήνα 2014. Διαθέσιμο από τη δικτυακή διεύθυνση: [ocp.teiath.gr](https://ocp.teiath.gr/).

**Σημείωμα Αδειοδότησης**

Το παρόν υλικό διατίθεται με τους όρους της άδειας χρήσης Creative Commons Αναφορά, Μη Εμπορική Χρήση Παρόμοια Διανομή 4.0 [1] ή μεταγενέστερη, Διεθνής Έκδοση. Εξαιρούνται τα αυτοτελή έργα τρίτων π.χ. φωτογραφίες, διαγράμματα κ.λ.π., τα οποία εμπεριέχονται σε αυτό. Οι όροι χρήσης των έργων τρίτων επεξηγούνται στη διαφάνεια «Επεξήγηση όρων χρήσης έργων τρίτων».

Τα έργα για τα οποία έχει ζητηθεί άδεια αναφέρονται στο «Σημείωμα Χρήσης Έργων Τρίτων».



[1] http://creativecommons.org/licenses/by-nc-sa/4.0/

Ως **Μη Εμπορική** ορίζεται η χρήση:

* που δεν περιλαμβάνει άμεσο ή έμμεσο οικονομικό όφελος από την χρήση του έργου, για το διανομέα του έργου και αδειοδόχο
* που δεν περιλαμβάνει οικονομική συναλλαγή ως προϋπόθεση για τη χρήση ή πρόσβαση στο έργο
* που δεν προσπορίζει στο διανομέα του έργου και αδειοδόχο έμμεσο οικονομικό όφελος (π.χ. διαφημίσεις) από την προβολή του έργου σε διαδικτυακό τόπο

Ο δικαιούχος μπορεί να παρέχει στον αδειοδόχο ξεχωριστή άδεια να χρησιμοποιεί το έργο για εμπορική χρήση, εφόσον αυτό του ζητηθεί.

**Επεξήγηση όρων χρήσης έργων τρίτων**

|  |  |
| --- | --- |
| © | Δεν επιτρέπεται η επαναχρησιμοποίηση του έργου, παρά μόνο εάν ζητηθεί εκ νέου άδεια από το δημιουργό. |
| διαθέσιμο με άδεια CC-BY | Επιτρέπεται η επαναχρησιμοποίηση του έργου και η δημιουργία παραγώγων αυτού με απλή αναφορά του δημιουργού. |
| διαθέσιμο με άδεια CC-BY-SA | Επιτρέπεται η επαναχρησιμοποίηση του έργου με αναφορά του δημιουργού, και διάθεση του έργου ή του παράγωγου αυτού με την ίδια άδεια. |
| διαθέσιμο με άδεια CC-BY-ND | Επιτρέπεται η επαναχρησιμοποίηση του έργου με αναφορά του δημιουργού. Δεν επιτρέπεται η δημιουργία παραγώγων του έργου. |
| διαθέσιμο με άδεια CC-BY-NC | Επιτρέπεται η επαναχρησιμοποίηση του έργου με αναφορά του δημιουργού. Δεν επιτρέπεται η εμπορική χρήση του έργου. |
| διαθέσιμο με άδεια CC-BY-NC-SA | Επιτρέπεται η επαναχρησιμοποίηση του έργου με αναφορά του δημιουργού και διάθεση του έργου ή του παράγωγου αυτού με την ίδια άδεια. Δεν επιτρέπεται η εμπορική χρήση του έργου. |
| διαθέσιμο με άδεια CC-BY-NC-ND | Επιτρέπεται η επαναχρησιμοποίηση του έργου με αναφορά του δημιουργού. Δεν επιτρέπεται η εμπορική χρήση του έργου και η δημιουργία παραγώγων του. |
| διαθέσιμο με άδεια CC0 Public Domain | Επιτρέπεται η επαναχρησιμοποίηση του έργου, η δημιουργία παραγώγων αυτού και η εμπορική του χρήση, χωρίς αναφορά του δημιουργού. |
| διαθέσιμο ως κοινό κτήμα | Επιτρέπεται η επαναχρησιμοποίηση του έργου, η δημιουργία παραγώγων αυτού και η εμπορική του χρήση, χωρίς αναφορά του δημιουργού. |
| χωρίς σήμανση | Συνήθως δεν επιτρέπεται η επαναχρησιμοποίηση του έργου. |

**Διατήρηση Σημειωμάτων**

* Οποιαδήποτε αναπαραγωγή ή διασκευή του υλικού θα πρέπει να συμπεριλαμβάνει:
* το Σημείωμα Αναφοράς
* το Σημείωμα Αδειοδότησης
* τη δήλωση Διατήρησης Σημειωμάτων
* το Σημείωμα Χρήσης Έργων Τρίτων (εφόσον υπάρχει) μαζί με τους συνοδευόμενους υπερσυνδέσμους.