

# CUPRINS

<b>INTRODUCERE .....</b>	<b>2</b>
<b>SCENARIUL .....</b>	<b>3</b>
<b>ERD (DIAGRAMA ENTITATE RELAȚIE) .....</b>	<b>6</b>
<b>MAPARE .....</b>	<b>8</b>
<b>REALIZARE APLICAȚIE .....</b>	<b>10</b>
<b>BIBLIOGRAFIE .....</b>	<b>17</b>

# INTRODUCERE

**Aplicația TEATRU** este un produs soft care poate fi utilizat pentru gestiunea unui teatru. Am ales această temă pentru a veni în întâmpinarea nevoilor managerilor care conduc aceste unități.

Acest program evidențiază întreaga activitate a unei instituții de acest astfel, oferind posibilitatea de a stoca, modifica, șterge, vizualiza date despre actori, piese, săli de spectacole, angajați și cluburi partenere.

Aplicația **TEATRU** este realizată interactiv în Oracle Application Express. Oracle Application Express permite proiectarea, dezvoltarea și implementarea de aplicații frumoase, receptive, bazate pe baze de date, utilizând numai browserul Web.

Oracle Application Express oferă toate instrumentele necesare pentru a fi productiv: un IDE bazat pe browser modern, o structură intuitivă de drag and drop, un editor de cod sofisticat și un editor bogat de proprietăți. Indiferent dacă sunteți un dezvoltator începător sau cu experiență, Application Express minimizează complexitatea implicată în aplicațiile cu mai multe fațete.

În cadrul Oracle Application Express o mare parte din procesul de dezvoltare este gestionat pentru noi: securitatea, autentificarea, interacțiunile bazei de date, validarea intrărilor, gestionarea stării de sesiune și multe alte dependențe. Toate instrumentele necesare sunt disponibile într-o singură platformă extensibilă, care rulează ca parte a Oracle Database.

Astfel, cu Application Express, ne putem concentra asupra problemei, accentul punându-se pe modelarea bazei de date. Totul pornește cu elaborarea unui scenariu care indică cererile clientului legate de afacerea pentru care solicită dezvoltarea aplicației de baze de date. Din cadrul scenariului specialistul va extrage entitățile cu atributele, precum și relațiile dintre acestea. Urmează etapa de mapare, care constă în trecerea de la modelul conceptual la modelul fizic al bazei de date. Având structura tabelor se trece la implementarea lor în cadrul Oracle Application Express, ultimul pas fiind reprezentat de realizarea aplicației.

# SCENARIUL

Primul pas în realizarea aplicației îl reprezintă scrierea scenariului, care evidențiază elementele necesare afacerii. Conținutul scenariului poate fi vizualizat în ceea ce urmează.

## TEATRU

La un teatru este nevoie să se memoreze date despre personal, actori, piesele de teatru și sălile de spectacol și cluburile de teatru partenere.

Fiecărui angajat îi este repartizat un număr format din 4 cifre reprezentând numărul de înregistrare, dar și codul său de acces. De asemenea pentru fiecare se cunoaște numele, prenumele, domeniul de activitate, adresa și eventual numărul de telefon. Fiecare angajat poate lucra în mai multe domenii și de asemenea și la cluburile partenere.

Pentru actori codul de acces și numărul de înregistrare este format din 4 cifre, precedate de literele "act". Și pentru ei se cunoaște numele, prenumele, adresa și numărul de telefon, obligatoriu, dar și studiile (ultimul nivel de studii absolvit, liceu sau facultate) și experiența (ani lucrați). Fiecare actor poate juca în mai multe piese de teatru, cu roluri principale și secundare; fie în sălile de teatru, fie în sălile partenere.

Teatru are în componența sa 5 săli de spectacol interne, două în aer liber și alte săli pe care le administrează alături de diferite cluburi de teatru. Astfel pentru cele 7 săli proprii se memorează un număr unic format din 2 cifre, denumirea sălii, tipul: "inside" sau "outside", în funcție de unde se află sala, responsabilul de sală, numărul de locuri, prețul biletului pentru sala respectivă și opțional data deschiderii sălii.

Pentru fiecare dintre cluburile de teatru partenere, se memorează un cod unic din 3 cifre, denumirea clubului, directorul, directorul adjunct, numărul de actori ai clubului, sala administrată și opțional data fondării clubului.

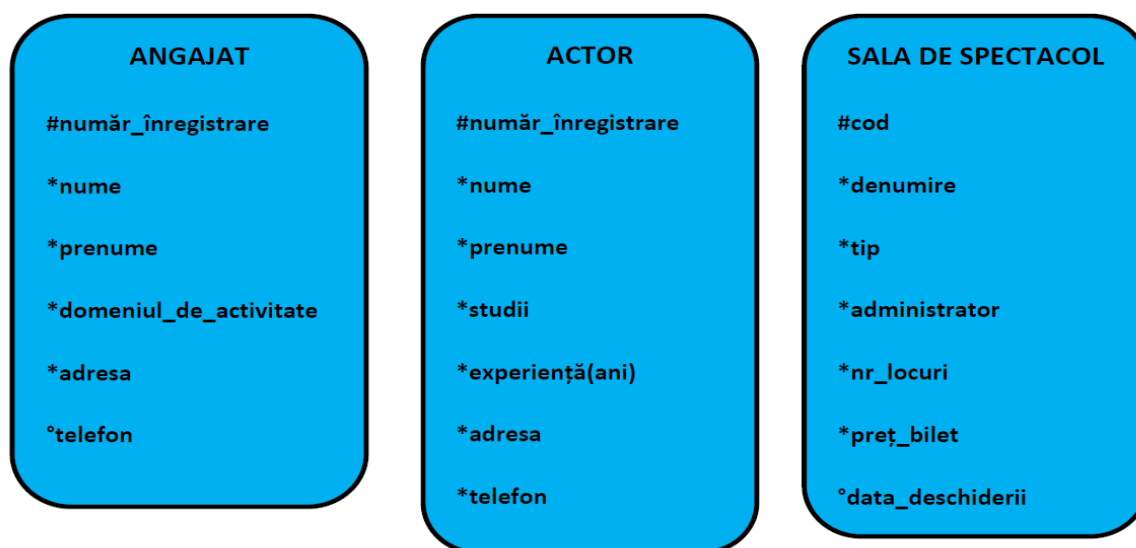
Pentru fiecare piesă de teatru în parte, se va memora un număr de înregistrare dat de directorul teatrului, numele piesei, regizorul și scenaristul, dacă se dorește, actorii principali, sala în care se va juca, genul piesei, iar după preferință durată și numărul de scene. Fiecare piesă se poate juca pe o singură scenă și de mai multe ori, iar distribuția poate suferi anumite modificări cerute de regizor sau de directorul teatrului.

# ENTITĂȚI

O entitate este un lucru, obiect, persoană sau eveniment care are semnificație pentru afacerea modelată, despre care trebuie să colectăm și să memorăm date. O entitate poate fi un lucru real, tangibil precum o clădire, o persoană, poate fi o activitate precum o programare sau o operație, sau poate fi o noțiune abstractă.

O entitate este reprezentată în ERD printr-un dreptunghi cu colțurile rotunjite. Numele entității este întotdeauna un substantiv la singular și se scrie în partea de sus a dreptunghiului cu majuscule.

O entitate este de fapt o clasă de obiecte și pentru orice entitate există mai multe instanțe ale sale. O instanță a unei entități este un obiect, persoană, eveniment, particular din clasa de obiecte care formează entitatea.



După cum se vede pentru a preciza o instanță a unei entități, trebuie să specificăm unele caracteristici ale acestui obiect, să-l descriem. Așadar, după ce am identificat entitățile trebuie să descriem aceste entități în termeni reali, adică să le stabilim atributele. Un atribut este orice detaliu care servește la identificarea, clasificarea, cuantificarea, sau exprimarea stării unei instanțe a unei entități. Atributele sunt informații specifice ce trebuie cunoscute și memorate.

Un atribut poate fi obligatoriu sau opțional. Dacă un atribut este obligatoriu, pentru



fiecare instanță a entității respective trebuie să avem o valoare pentru acel atribut. Pentru un atribut opțional putem avea instanțe pentru care nu cunoaștem valoarea atributului respectiv. Un atribut obligatoriu este precedat în ERD de un asterisc \*, iar un atribut opțional va fi precedat de un cerculeț o.

Atributele care definesc în mod unic instanțele unei entități se numesc identificator unic (UID). Atributele care fac parte din identificatorul unic al unei entități vor fi precedate de semnul diez #. Atributele din UID sunt întotdeauna obligatorii, însă semnul # este suficient, nu mai trebuie pus și un semn asterisc în fața acestor atribute.

În cele două imagini sunt evidențiate entitățile și atributele specifice aplicației Teatru.

# ERD

## (Diagrama Entitate Relație)

În lumea reală, obiectele nu există izolat. Între ele există relații. Așadar, după ce au fost identificate entitățile și atributele acestor entități este timpul să fie puse în evidență relațiile care există între aceste entități, modul în care acestea comunică între ele. O relație este o asociere, legătură, sau conexiune existentă între entități și care are o semnificație pentru afacerea modelată. Orice relație este bidirecțională, legând două entități sau o entitate cu ea însăși.

Orice relație este caracterizată de următoarele elemente:

1. numele relației ; 2. opționalitatea relației; 3. gradul (cardinalitatea) relației.

În cadrul diagramei entitate-relație, o relație va fi reprezentată printr-o linie ce unește cele două entități. Deoarece o relație este bidirecțională, linia ce unește cele două entități este compusă din două segmente distincte, câte una pentru fiecare entitate. Tipul segmentului care pleacă de la o entitate ne va indica opționalitatea relației dintre această entitate și entitatea aflată în cealaltă parte a relației. Dacă acest segment este continuu este vorba de o relație obligatorie, o linie întreruptă indică o relație opțională.

Modul în care o linie se termină spre o entitate este important. Dacă se termină printr-o linie simplă, înseamnă că o instanță și numai una a acestei entități este în relație cu o instanță a celeilalte entități. Dacă linia se termină cu trei linii (picior de cioară) înseamnă că mai multe instanțe ale entității pot corespunde unei instanțe a celeilalte entități.

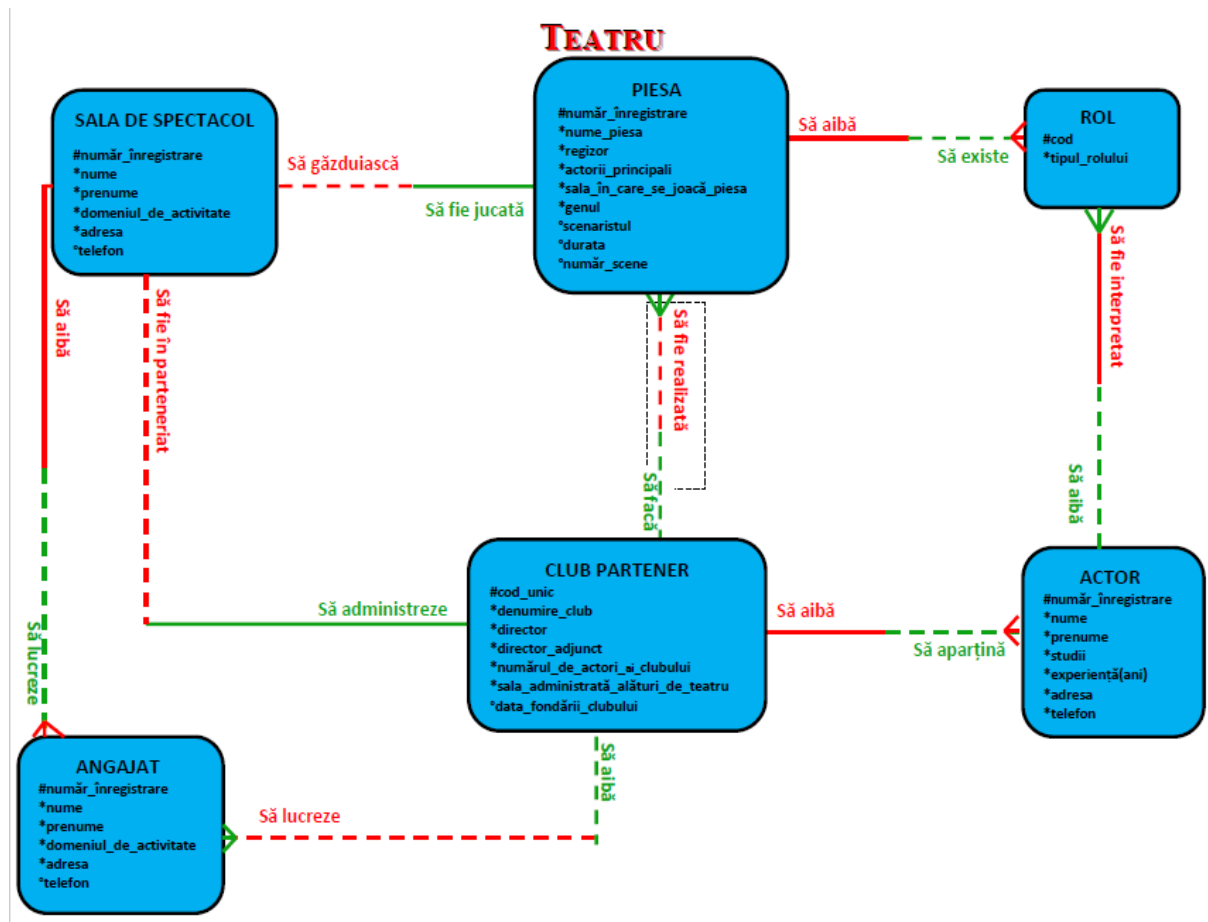
Caracteristica relației	Valoare	Mod de reprezentare
<b>Numele relației</b>	un verb	se scrie deasupra relației
<b>Opționalitatea</b>	relație obligatorie (TREBUIE)	linie continuă —————
	relație opțională (POATE)	linie întreruptă -----
<b>Cardinalitatea</b>	una și numai una	linie simplă ———
	una sau mai multe	picior de cioară <=

Variantele de relații care pot exista între două entități sunt prezentate mai jos:

- relații one-to-one – acest tip de relație este destul de rar întâlnit. Uneori astfel de relații pot fi modelate transformând una dintre entități în atribut al celeilalte entități.

- relații one-to-many – sunt cele mai întâlnite tipuri de relații
- relații many-to-many – aceste tipuri de relații apar în prima fază a proiectării bazei de date, însă ele trebuie să fie ulterior eliminate.

Diagrama Entitate-Relație corespunzătoare aplicației Teatru este evidențiată în cele ce urmează.



# MAPARE

Transformarea modelului conceptual, a ERD-ului, în modelul fizic, adică în baza de date propriu zisă, se numește mapare. Acest proces implică transformarea fiecărui element al ERD-ului, după cum urmează: Entități → tabele, attribute → câmpuri, coloane, UID → cheie primară, relație → cheie străină, business rules → constrângeri.

În general, la maparea unei relații de tip one-to-many, vom introduce în tabela corespunzătoare entității de pe partea many a relației cheia primară a entității de pe partea one a relației. Câmpurile astfel introduse se vor numi cheie străină (foreign key). Așadar: - cheia străină a unei tabele este cheia primară din tabela referință - cheia străină este întotdeauna introdusă în tabela corespunzătoare entității din partea many a relației.

Dându-se două entități A și B legate între ele printr-o relație one-to-one, este evident că putem include cheia primară A în cadrul tabelii B, dar putem proceda la fel de bine și invers, incluzând cheia primară a tabelii B în cadrul tabelii A, deoarece fiecărei instanțe a entității A îi corespunde cel mult o instanță a entității B, dar și invers, oricărei instanțe a entității B îi corespunde cel mult o instanță a entității A. Decizia depinde de specificul afacerii modelate.

Maparea corespunzătoare aplicației Teatru poate fi urmărită mai jos.

<b>PIESE (PSA)</b>		
<b>Tipul cheii</b>	<b>Opționalitatea</b>	<b>Numele coloanei</b>
pk	*	nr_inreg
	*	nume_piesa
	*	regizor
	*	gen
	o	scenarist
	o	durata
	o	nr_scene
fk	*	ssl_cod
fk	o	cpr_cod

<b>ROLURI (ROL)</b>		
<b>Tipul cheii</b>	<b>Opționalitatea</b>	<b>Numele coloanei</b>
pk	*	cod
	*	tip_rol
fk	o	psa_nr_in
fk	*	atr_nr_in



<b>ACTORI (ATR)</b>		
<b>Tipul cheii</b>	<b>Opționalitatea</b>	<b>Numele coloanei</b>
pk	*	nr_inreg
	*	nume
	*	prenume
	*	studii
	*	experienta
	*	adresa
	*	telefon
fk	o	cpr_cod

<b>CLUBURI PARTENERE (CPR)</b>		
<b>Tipul cheii</b>	<b>Opționalitatea</b>	<b>Numele coloanei</b>
pk	*	cod_unic
	*	denumire
	*	director
	*	dir_adj
	o	d_fondare
fk	*	ssl_cod

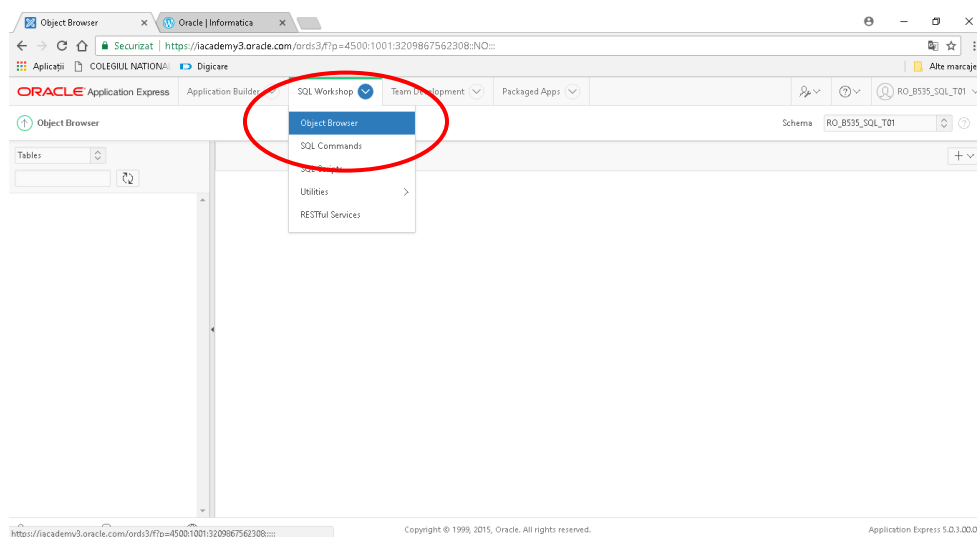
<b>SALI SPECTACOLE (SSL)</b>		
<b>Tipul cheii</b>	<b>Opționalitatea</b>	<b>Numele coloanei</b>
pk	*	cod
	*	denumire
	*	tip
	*	admin
	*	nr_locuri
	*	pret_bilet
	o	d_deschid

<b>ANGAJATI (AGT)</b>		
<b>Tipul cheii</b>	<b>Opționalitatea</b>	<b>Numele coloanei</b>
pk	*	nr_inreg
	*	nume
	*	prenume
	*	dom_activ
	*	adresa
	o	telefon
fk	o	ssl_cod
fk	o	cpr_cod

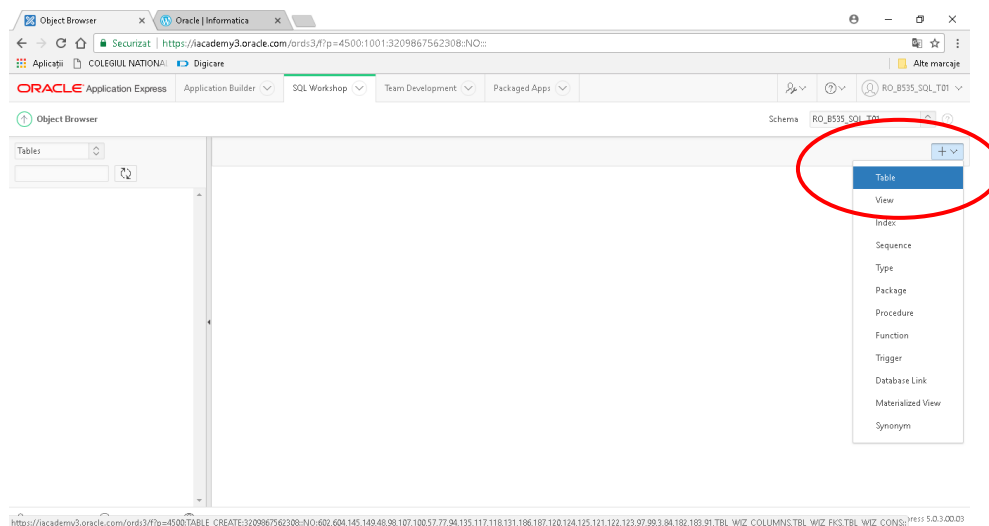
# REALIZARE APLICATIE

## I.Realizare tabele

1.Click SQL Workshop – se alege opțiunea Object Browser



2.Click pe butonul Create – se alege opțiunea Table

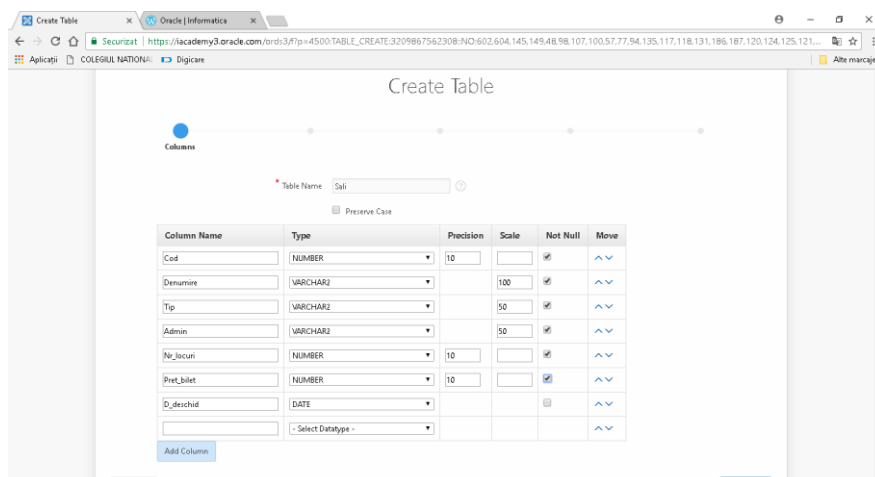


3.Se scrie numele tabelului

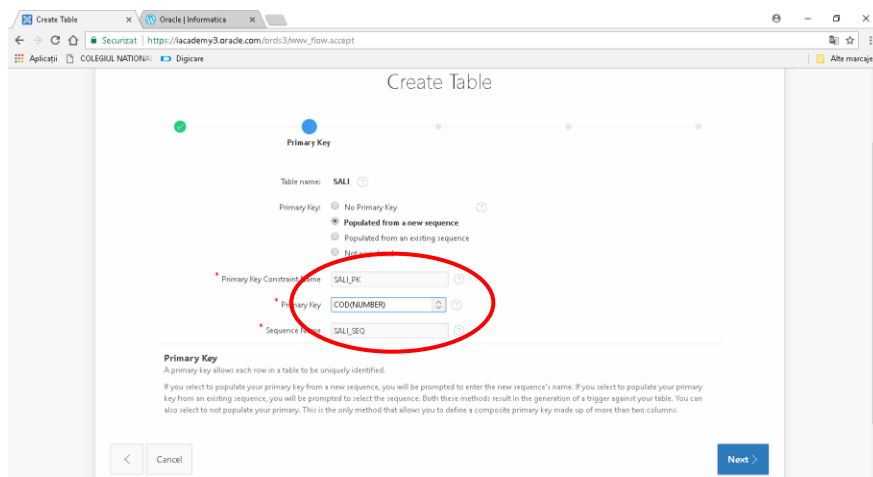
4.Se completează numele coloanelor (inclusiv cheile străine)

5.Se stabilesc tipurile de date pentru coloane

6.Se acționează butonul NEXT



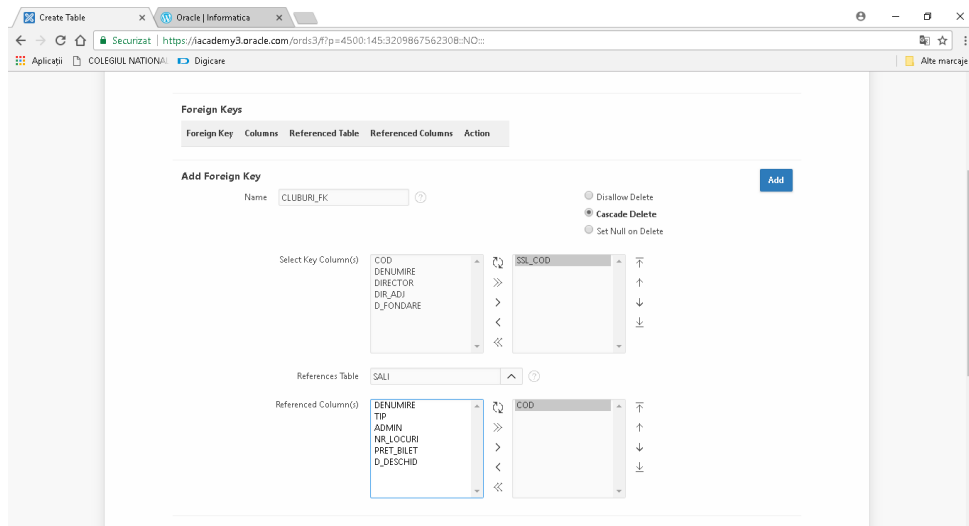
7. Se stabilește cheia primară – se alege a doua opțiune (Populated from a new sequence)  
 - la Primary Key se alege coloana dorită (cea care va fi cheia primară)



8. Se acționează butonul NEXT

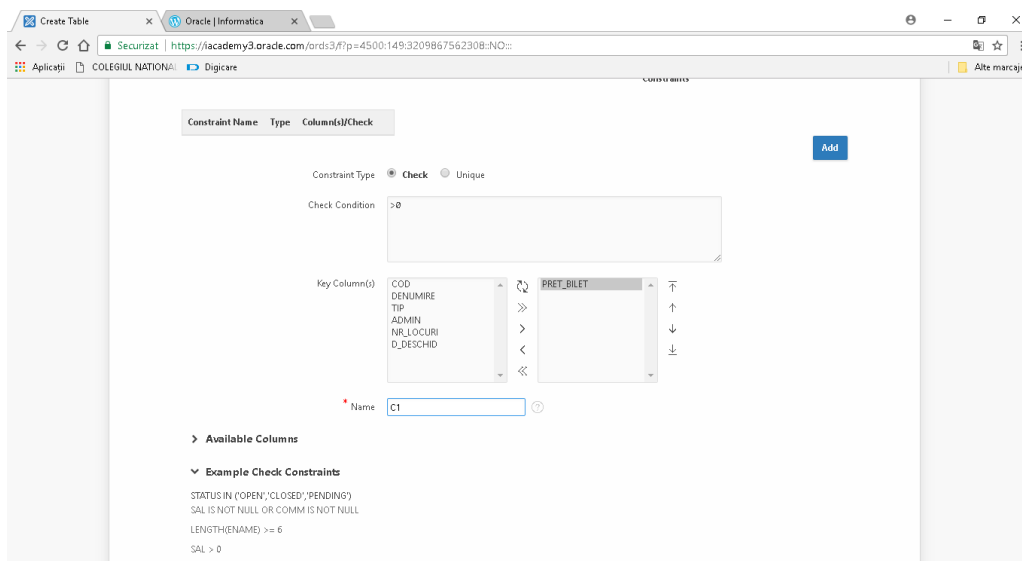
9. Se stabilesc cheile străine, dacă există

- se alege una dintre opțiunile Disallow Delete, Cascade Delete, Set Null on Delete în funcție de dorința utilizatorului
- se alege cheia străină din tabel
- se alege tabelul cu care este în relație
- se selectează cheia primară din tabelul ales
- se folosește butonul Add pentru adăugarea a mai multor chei străine



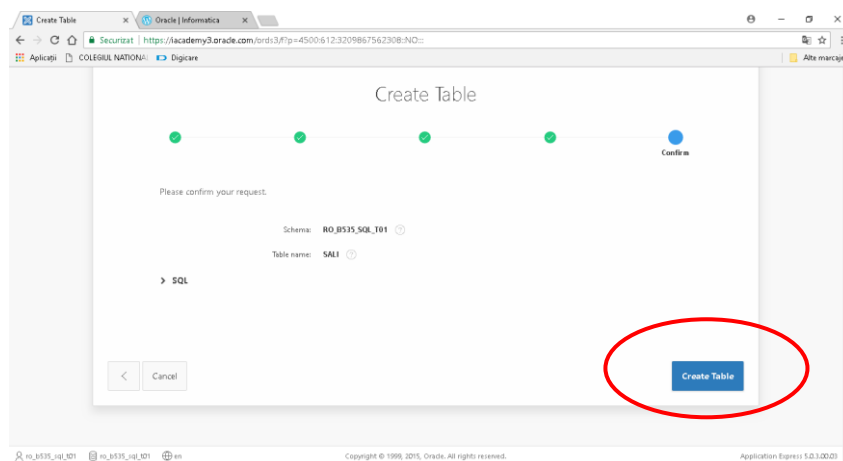
10. Se acționează butonul NEXT

11. Se stabilesc constrângeri, dacă există. Pentru mai multe constrângeri se utilizează butonul Add.



12. Se acționează butonul NEXT

13. Se acționează butonul Create Table

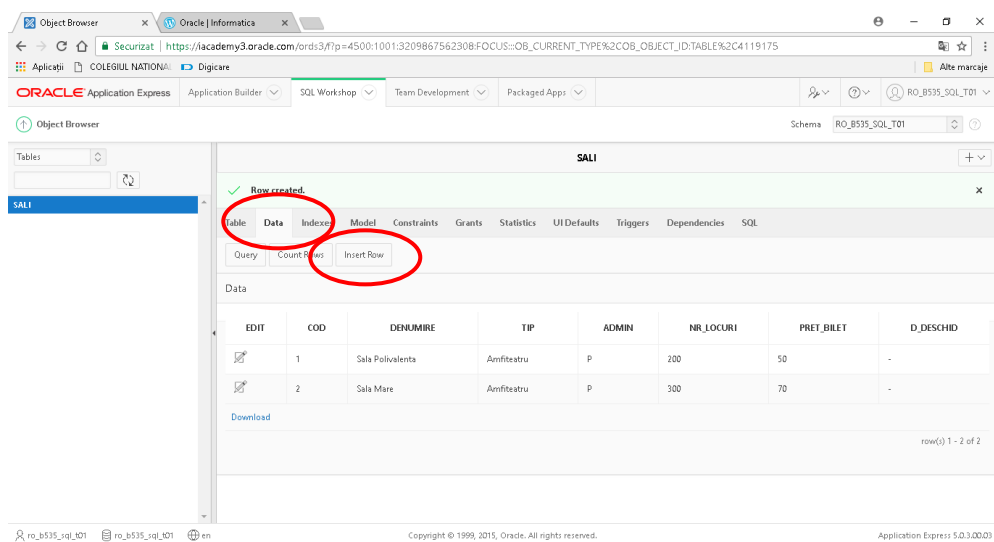


Se procedează la fel cu celelalte tabele

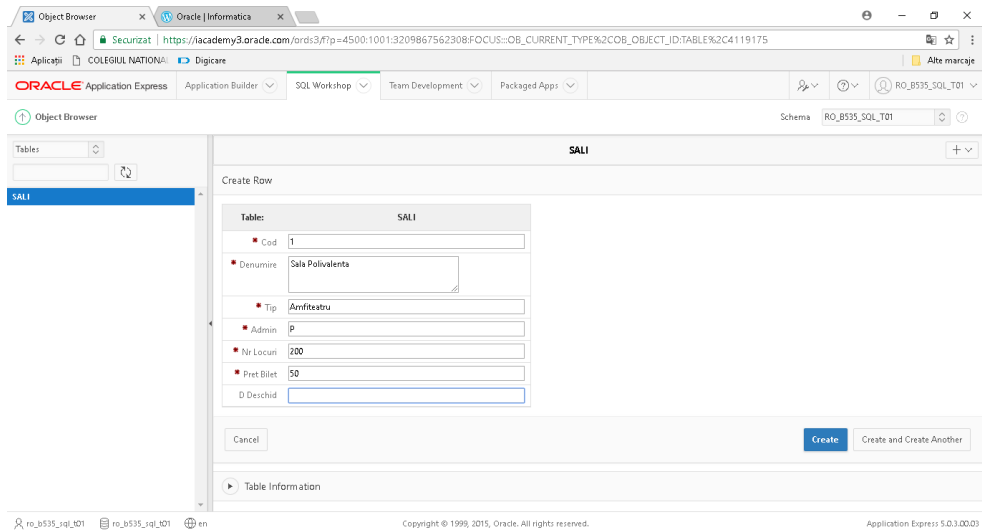
## II. Popularea cu date a tabelor

1. Click pe Data

2. Click pe Insert Row



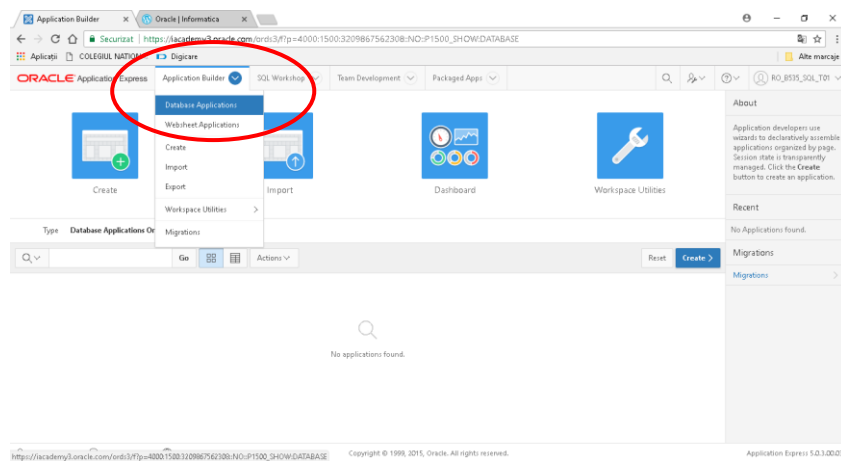
### 3. Se introduc datele



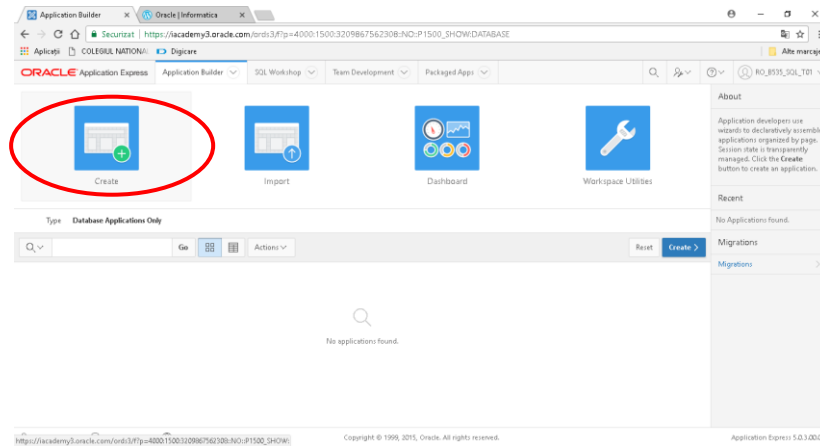
4. Se acționează butonul Create pentru a adauga rândul în tabel și pentru a opri inserarea sau se acționează butonul Create and Create Another pentru a adauga rândul în tabel și pentru a continua adaugarea altor rânduri.

## II. Realizare aplicație

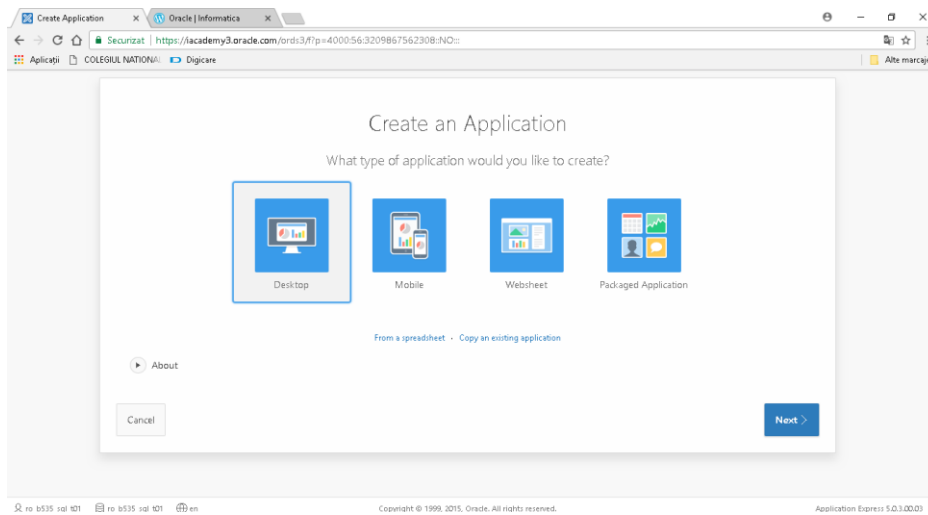
### 1. Click Application Builder – Database Applications



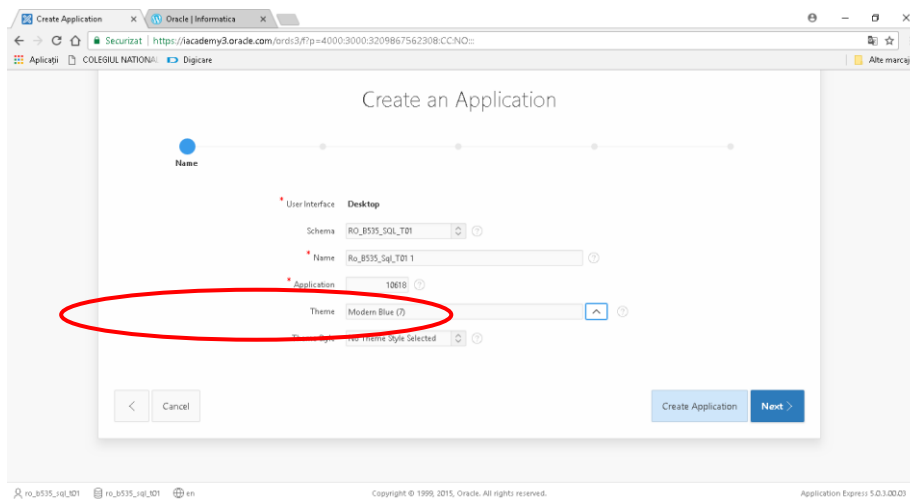
## 2. Click Create



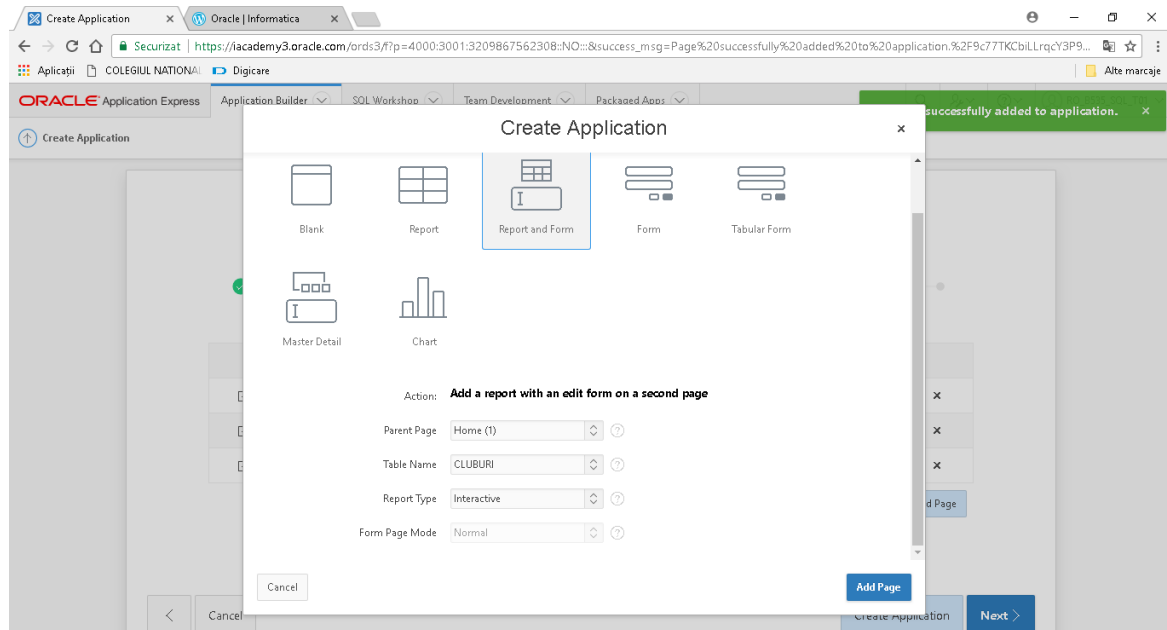
## 3. Se alege tipul dorit pentru aplicație, apoi se acționează butonul Next



## 4. Se alege tema dorită pentru aplicație, apoi se acționează butonul Next (nu se modifică numărul aplicației – Application)



## 5. Se adaugă paginile dorite



6. Se aleg componentele partajate, dacă există, apoi se acționează butonul Next

7. Se stabilesc atributele pentru aplicație, apoi se acționează butonul Next

8. Se acționează butonul Create Application

9. Se acționează butonul Run Application pentru a rula aplicația



# BIBLIOGRAFIE

1. Popescu Carmen – *Manual de Informatică pentru clasa a XII-a*, Editura L&S Informat, 2007
2. Cursul Oracle – Database Design
3. <http://info12.wikispaces.com/>
4. <https://iacademy3.oracle.com>