3.3. Duomenų vientisumo sąlygos

Taisyklė apibrėžianti tam tikrus suvaržymus duomenims dar vadinama **duomenų vientisumo sąlyga**.

Aptarsime tris jų:
- kategorijų vientisumas;
- nuorodų vientisumas;
- funkcinės priklausomybės.

Realaus pasaulio objektai reliacinėje teorijoje vaizduojami lentelės eilutėmis ir vadinami **kategorijomis**.

Kategorijų vientisumo taisyklė: joks lentelės rakto atributas nei vienoje eilutėje negali turėti `NULL` reikšmę.

Nuorodų vientisumo taisyklė: kiekvieno išorinio rakto reikšmė turi būti arba tuščia, arba sutapti su viena pirminio rakto reikšme lentelėje, į kurią nurodo išorinis raktas.

Funkcines priklausomybes aptarsime vėliau.

3.4. Duomenų anomalijos

Tarkime, vietoje dviejų lentelių *Vykdymas* ir *Projektai* turime vieną - *Projektai_Vykdymas*.

DB ***Darbai***:

Vykdytojai (Nr, Pavardė, Kvalifikacija, Kategorija, Išsilavinimas)

Projektai_Vykdymas(Projektas, Pavadinimas, Svarba, Trukmė, Pradžia, Vykdytojas, Statusas, Valandos)

Išorinis raktas: *Vykdytojas* nukreipia į *Vykdytojai*

Užpildome lentelę *Projektai_Vykdymas* ankstesniųjų dviejų lentelių *Projektai* ir *Vykdymas* duomenimis:

Projektai_Vykdymas

Projektas	*Pavadinimas*	*Svarba*	*Trukmė*	*Vykdytojas*	…
1	Studentų apskaita	Maža	12	1	
1	Studentų apskaita	Maža	12	2	
1	Studentų apskaita	Maža	12	3	
1	Studentų apskaita	Maža	12	4	
2	Buhalterinė apskaita	Vidutinė	10	1	
2	Buhalterinė apskaita	Vidutinė	10	2	
2	Buhalterinė apskaita	Vidutinė	10	4	
3	WWW svetainė	Didelė	6	1	
3	WWW svetainė	Didelė	6	2	
3	WWW svetainė	Didelė	6	3	

Pertekliniai (pasikartojantys) **duomenys** ne tik užima vietą atmintyje, bet gali būti duomenų prieštaringumo priežastimi.

Pailginame projekto Nr. 1 trukmę

Projektas	*Pavadinimas*	*Svarba*	*Trukmė*	*Vykdytojas*	..
1	Studentų apskaita	Maža	**15**	1	
1	Studentų apskaita	Maža	12	2	
1	Studentų apskaita	Maža	12	3	
1	Studentų apskaita	Maža	12	4	
2	Buhalterinė apskaita	Vidutinė	10	1	
..	...	...	...	..	

Atnaujinimo anomalija–duomenų **prieštaringumas**, atsirandantis dėl duomenų pertekliaus, atnaujinus tik dalį jų (ne visas kopijas).

Vykdytojai Nr. 1,2,3 išeina iš darbo:

Projektas	*Pavadinimas*	*Svarba*	*Trukmė*	*Vykdytojas*	…
1	**Studentų apskaita**	**Maža**	**12**	**1**	
1	**Studentų apskaita**	**Maža**	**12**	**2**	
1	**Studentų apskaita**	**Maža**	**12**	**3**	
1	Studentų apskaita	Maža	12	4	
2	**Buhalterinė apskaita**	**Vidutinė**	**10**	**1**	
2	**Buhalterinė apskaita**	**Vidutinė**	**10**	**2**	
2	Buhalterinė apskaita	Vidutinė	10	4	
3	**WWW svetainė**	**Didelė**	**6**	**1**	
3	**WWW svetainė**	**Didelė**	**6**	**2**	
3	**WWW svetainė**	**Didelė**	**6**	**3**	

Negalime pašalinti duomenų apie vykdytojus Nr. 1, 2 ir 3, neištrindami ir duomenų apie projektą Nr. 3.

Šalinimo (trynimo) anomalija

– nenumatytas reikalingų duomenų praradimas, susijęs su kitų duomenų šalinimu.

Užregistruokime naują projektą:

'Paskaitų tvarkaraštis'

- Reikia įvesti į *Projektai_Vykdymas* naują eilutę
- *Vykdytojas* yra rakto dalis ⇒ `NOT NULL`
- Negalime įvesti eilutės nepriskyrus reikšmės *Vykdytojas*

⇒ Negalime užregistruoti projekto, kol nepaskirtas bent vienas vykdytojas.

Įvedimo anomalija – negalėjimas įvesti duomenų, dėl kitų duomenų nebuvimo, t.y. neįvedant ir kitų, tiesiogiai nesusijusių duomenų.

Kad šių anomalijų nebūtų,
vietoje *Projektai_Vykdymas* turi būti 2 lentelės

- *Projektai*
- *Vykdymas*

Duomenų dubliavimo ir anomalijų išvengiama skaidant lenteles.

Lentelių skaidymas - lentelės padalinimas į kelias lenteles, siekiant išvengti duomenų dubliavimo ir neprarasti duomenų vientisumo.

3.5. Pirmoji norminė forma

Normine forma (NF) vadinamos sąlygos, kurias turi tenkinti DB reliacinė schema, kad būtų išvengta tam tikrų nepageidaujamų savybių.

Lentelė yra **pirmos norminės formos** (1NF), jei visų jos atributų reikšmės yra **atomai**.

Reikšmė yra atomas, jei ji nėra nei aibė, nei sąrašas.

Lentelei *Projektai_Vykdymas* būdingos duomenų anomalijos, kurios pašalinamos, skaidant lentelę į dvi.

Problemų lentelėje *Projektai_Vykdytojai* **priežastis** - duomenų perteklius.

Problemų **sprendimo būdas** – lentelės skaidymas.

Pakeiskime blogai sudarytą lentelę *Projektai_Vykdymas* kita lentele:

Projektai_Vykdytojai (*<u>Projektas</u>, Pavadinimas, Svarba, Trukmė, Pradžia, Vykdytojai, Statusas, Valandos*)

Projektai_Vykdytojai

Projektas	*Pavadinimas*	*Svarba*	*Trukmė*	*Vykdytojai*	...
1	Studentų apskaita	Maža	12	{1,2,3,4}	
2	Buhalterinė apskaita	Vidutinė	10	{1,2,4}	
3	WWW svetainė	Didelė	6	{1,2,3}	

Šioje lentelėje nėra duomenų dubliavimo, kuris buvo lentelėje *Projektai_Vykdymas*.

Šiai lentelei **negalima apibrėžti išorinio rakto**, todėl galimas nuorodų vientisumo pažeidimas.

Projektai_Vykdytojai nėra 1NF.

Projektai_Vykdymas yra 1NF, tačiau jos schema irgi **nėra** gera.

Išvada: 1NF yra per silpnas reikalavimas.

3.6. Funkcinės priklausomybės (FP)

FP – svarbiausia apribojimų rūšis.

Jei atributų aibės *A* reikšmės korteže vienareikšmiškai apibrėžia atributų aibės *B* reikšmes korteže, tai **priklausomybę** tarp *A* ir *B* **vadiname funkcine** (FP).

Jei *A* ir *B* yra *L* atributų aibės, tai $A \rightarrow B$ reiškia: jei dvi lentelės *L* eilutės (kortežai) turi vienodas *A* reikšmes, tai *B* reikšmės taip pat sutampa.

$A \rightarrow B$: *B* **f-priklauso** nuo *A*, *A* **f-apibrėžia** *B*.

FP kairiosios dalies atributai vadinami **determinantu**.

Projektai_Vykdymas (*<u>Projektas</u>, Pavadinimas, Svarba, Trukmė, Pradžia, <u>Vykdytojas</u>, Statusas, Valandos*)

Atributų tarpusavio FP:

Projektas $\rightarrow$ *Pavadinimas*
Projektas $\rightarrow$ *Svarba*
Projektas $\rightarrow$ *Trukmė*
Projektas $\rightarrow$ *Pradžia*

trumpiau:

Projektas $\rightarrow$ {*Pavadinimas, Svarba, Trukmė, Pradžia*}

{*Projektas, Vykdytojas*} $\rightarrow$ {*Statusas, Valandos*}

FP apibendrina sąvoką „lentelės viršraktis".

$L(R)$ **viršraktis** – tai lentelės visų atributų aibės R poaibis S ($S \subseteq R$):

$$S \rightarrow R$$

FP, kaip ir kiti apribojimai, yra teiginys apie reliacinę schemą, o ne apie konkrečius lentelės duomenis.

Pvz., tik esamiems duomenims:

Svarba $\rightarrow$ *Trukmė*

FP $A_1, A_2, \ldots, A_n \to B_1, B_2, \ldots, B_m$:

- **trivialioji**, jei

$$\{B_1, B_2, \ldots, B_m\} \subseteq \{A_1, A_2, \ldots, A_n\};$$

- **netrivialioji**, jei

$$\exists B_i \in \{B_1, B_2, \ldots, B_m\} \text{ ir } B_i \notin \{A_1, A_2, \ldots, A_m\};$$

- **visiškai netrivialioji**, jei

$$\forall i : 1, \ldots, m : B_i \notin \{A_1, A_2, \ldots, A_m\}.$$

Algoritmas Ar lentelė L šiuo metu tenkina FP $A \to B$?

1 žingsnis. L eilutes sugrupuojame pagal atributų A reikšmes taip, kad eilutės su vienodomis A reikšmėmis priklausytų tai pačiai grupei.

2 žingsnis. **Jei** kiekvienoje grupėje atributų B reikšmės taip pat sutampa, **tai** L tenkina FP $A \to B$, **kitaip** netenkina.

3.7. FP uždarinys

Vienas FP kartais galima pakeisti kitomis.

Tarkime, lentelėje $L(A, B, C)$ galioja FP:

$$A \to B$$

$$B \to C$$

Tuomet lentelėje L galios ir $A \to C$.

Aibė visų galimų FP, kurios apibrėžiamos FP aibe F, vadinama **aibės F uždariniu** ir žymima F^+.

W.W. Armstrong suformulavo išvedimo taisykles (aksiomas), kuriomis galima gauti visas aibės F^+ FP.

Armstrongo aksiomos. Tarkime, A, B ir C yra lentelės $L(R)$ atributų aibės R poaibiai. Žymėsime $AB \equiv A \cup B$.

1. Refleksyvumas: jei $B \subseteq A$, tai $A \to B$
2. Papildymas: jei $A \to B$, tai $AC \to BC$
3. Tranzityvumas: jei $A \to B$ ir $B \to C$, tai $A \to C$

Visos šios taisyklės gali būti įrodytos iš FP apibrėžimo.

Ši aibė yra pilnoji, t.y. duotai FP aibei F, visos galiojančios FP gali būti išvestos panaudojant tik šias taisykles.

Paprastumo dėlei dar naudojama:

4. Apsibrėžtis: $A \to A$
5. Dekompozicija: jei $A \to BC$, tai $A \to B$ ir $A \to C$
6. Apjungimas: jei $A \to B$ ir $A \to C$, tai $A \to BC$.
7. Kompozicija: jei $A \to B$ ir $C \to D$, tai $AC \to BD$, čia $D \subset R$.

Iš lentelėje galiojančių FP aibės, randami lentelės raktai.

Aibė visų atributų, kurie f-priklauso nuo atributų S, yra **atributų aibės S uždarinys** S^+ FP aibės F atžvilgiu.

S yra **viršraktis**, jei $S \to R$, arba $S^+ = R$.

Pavyzdys

$R(A, B, C, D)$ – lentelė

1) $A \to B$; 2) $D \to A$; 3) $C \to D$;

? $C \to ABCD$

(4) $C \to C$ – pagal apsibrėžties taisyklę

(5) $C \to A$ – iš (3,2) pagal tranzityvumo t.

(6) $C \to B$ – iš (5,1) pagal tranzityvumo t.

(7) $C \to ABCD$ – iš (3,4,5,6) pagal jungimo t.

$\Rightarrow C$ – R raktas

Algoritmas atributų aibės S uždariniui S^+ FP F atžvilgiu rasti.

```
S+ := S
repeat
   T := S+
   for each X → Y ∈ F
      if X ⊆ S+ then S+ := S+ ∪ Y
   endfor
until (T = S+)
```

FP aibei F: $X \to Y \in F^+$ tada ir tik tada, kai $Y \subseteq X^+$.

Pavyzdys. $R(A, B, C, D)$ – lentelė

1) $A \to B$; 2) $D \to A$; 3) $C \to D$;

? C^+

$C^+ := C$;

1-oji peržiūra $C^+ := CD$ – (3)

2-oji peržiūra $C^+ := CDA$ – (2)

3-oji peržiūra $C^+ := CDAB$ – (1)

4-oji peržiūra $C^+ := CDAB$ – rezultatas

$\Rightarrow C^+ := CDAB$

Iš vienų FP galime išvesti kitas.
Kada dvi FP aibės aprašo tas pačias savybes?
Dvi FP aibės F ir G yra **ekvivalenčios**, jei $F^+ = G^+$.

Kad patikrinti, ar $F^+ \equiv G^+$ (yra **ekvivalenčios**), pakanka patikrinti,

- ar $\forall\ X \to Y \in F$ yra išvedama aibėje G, t.y. ar $X \to Y \in G^+$
- ar $\forall\ U \to V \in G$ yra išvedama aibėje F, t.y. ar $U \to V \in F^+$

Pavyzdys. FP aibės:

$$F = \{A \to BC;\ B \to C\}$$
$$G = \{A \to B;\ \ B \to C\}$$

? $F^+ \equiv G^+$

? $A \to BC \in G^+$

$A \to C \in G^+$ – pagal tranzityvumo taisyklę

$A \to BC \in G^+$ – pagal jungimo taisyklę

$\Rightarrow F^+ \subseteq G^+$

? $A \to B \in F^+$

$A \to B \in F^+$ – pagal dekompozicijos taisyklę

$\Rightarrow G^+ \subseteq F^+$

Lentelės $L(R)$, kurioje galioja FP aibė F, **raktas** yra jos atributų aibės R poaibis K ($K \subseteq R$), toks kad:

1) $K \to R$ (vienareikšmis tapatumas),
2) $\forall B \subset K : B \to R \notin F^+$ (pertekliaus nebuvimas).

Algoritmas lentelės $L(R)$ raktui K FP aibės F atžvilgiu rasti.

```
K := R ;
for each  A ∈ K
   T := (K – A)+ in F ;
   if  T = R  then K := K – A  endif;
endfor.
```

Taip surandamas 1 raktas.

Visiems raktams surasti yra taikomas sudėtingesnis algoritmas.

FP uždarinyje yra galimos perteklinės FP.
Tai gali labai apsunkinti FP savybių tyrimą.
Todėl ieškomas minimalus (standartinis) denginys.

FP aibė F vadinama **minimaliąja**, jeigu ji tenkina reikalavimus:

a) $\forall\ X \to Y \in F$, Y yra sudaryta tik iš 1 atributo;

b) nėra FP, kurią pašalinus, gautume tapačią (ekvivalenčią) aibę;

c) nėra $X \to A$, kurią pakeitus į $Y \to A$, $Y \subset X$, gautume ekvivalentišką aibę.

FP aibės F **minimaliu denginiu** vadinama: minimalioji FP aibė $F_{\min}$, kuri yra ekvivalenti aibei F.

FP aibei F gali egzistuoti kelios $F_{\min}$.

Visada galima surasti bent vieną jų.

Minimalusis denginys yra minimalus dviem aspektais:

- visos FP minimaliai paprastos – dešinėje 1 atributas, determinante nėra perteklinių atributų;
- nėra nereikalingų FP.

FP aibei F minimalųjį denginį $F_{\min}$ galima sudaryti **trimis žingsniais**:

1) visas FP perrašom (taikom dekompozicijos taisyklę) **standartiniu pavidalu**, kad dešinėje pusėje būtų tik 1 atributas;
2) pašalinam visas perteklines FP, be kurių likusi FP aibė yra tapati aibei F;
3) kiekvienai FP, tikrinam, ar determinante visi atributai yra būtini, – determinanto atributą ištriname, jei ir be jo FP aibė lieka tapati pradinei aibei F.

Algoritmas FP aibės F minimaliajam denginiui $F_{\min}$ rasti.

```
Fmin := F;
for each  X → A1 A2 ... An ∈ Fmin
  Fmin := Fmin – (X → A1 A2 ... An);
  for  i := 1 to n
    Fmin := Fmin ∪ X → Ai;
  endfor
endfor
```

```
for each X → A ∈ F_min
    T := X⁺ in (F_min − (X → A));
    if A ∈ T then
      F_min := F_min − (X → A)
    endif;
endfor
```



```
for each X → A ∈ F_min
  for each B ∈ X
    T := (X − B)⁺ in F_min ;
    if A ∈ T then
      F_min := F_min − (X → A);
      F_min := F_min ∪ ((X − B) → A);
    endif
  endfor
endfor
```


Pvz. $F := AB \to CD,\ A \to B,\ B \to C$

1) $F_{min} := AB \to C, AB \to D, A \to B, B \to C$

2) $AB^+ = ABCD$ **in** $F_{min} - AB \to C \Rightarrow C \in AB^+$

$F_{min} := AB \to D, A \to B, B \to C$

3) $A^+ = ABCD$ **in** $F_{min} \Rightarrow D \in A^+$

taip pat ir **in** $F_{min} - (AB \to D) \cup (A \to D)$

$F_{min} := A \to D, A \to B, B \to C$