

Databáze pro basketbalovou ligu NBA

Dokumentace semestrální práce pro předmět 4IT218 Databáze

Jakub Hošek

2020/21 Zimní semestr

Uživatelské jméno: hosj03

1.	Popis	3
2.	Konceptuální schéma reality	4
3.	Konceptuální datový model	5
4.	Dokumentace databáze.....	6
4.1.	Fyzický datový model	6
4.2.	Definice relačních tabulek a souvisejících objektů	7
4.3.	Integritní omezení	15
4.3.1.	HRAC	15
4.3.2.	HRAJE_ZA.....	15
4.3.3.	SPONZOR	16
4.3.4.	SPONZORUJE	17
4.3.5.	TYM.....	18
4.3.6.	ZAPAS.....	18
4.4.	Definice přístupových práv.....	21
4.4.1.	Přiřazení přístupových práv pro uživatele STUDENT.....	21
4.4.2.	Přiřazení přístupových práv pro uživatele DB4IT218	21
4.5.	Definice dalších databázových objektů	22
4.5.1.	Sekvence čísel	22
4.5.2.	Funkce pro kontrolu zápasů	22
5.	Obsah databáze	24
5.1.	SQL příkazy pro naplnění databáze daty	24
5.2.	Opis vložených dat.....	26
5.2.1.	Tabulka HRAC	26
5.2.2.	Tabulka HRAJE_ZA	26
5.2.3.	Tabulka SPONZOR.....	26
5.2.4.	Tabulka SPONZORUJE.....	26
5.2.5.	Tabulka TYM	26
5.2.6.	Tabulka ZAPAS	27

1. Popis

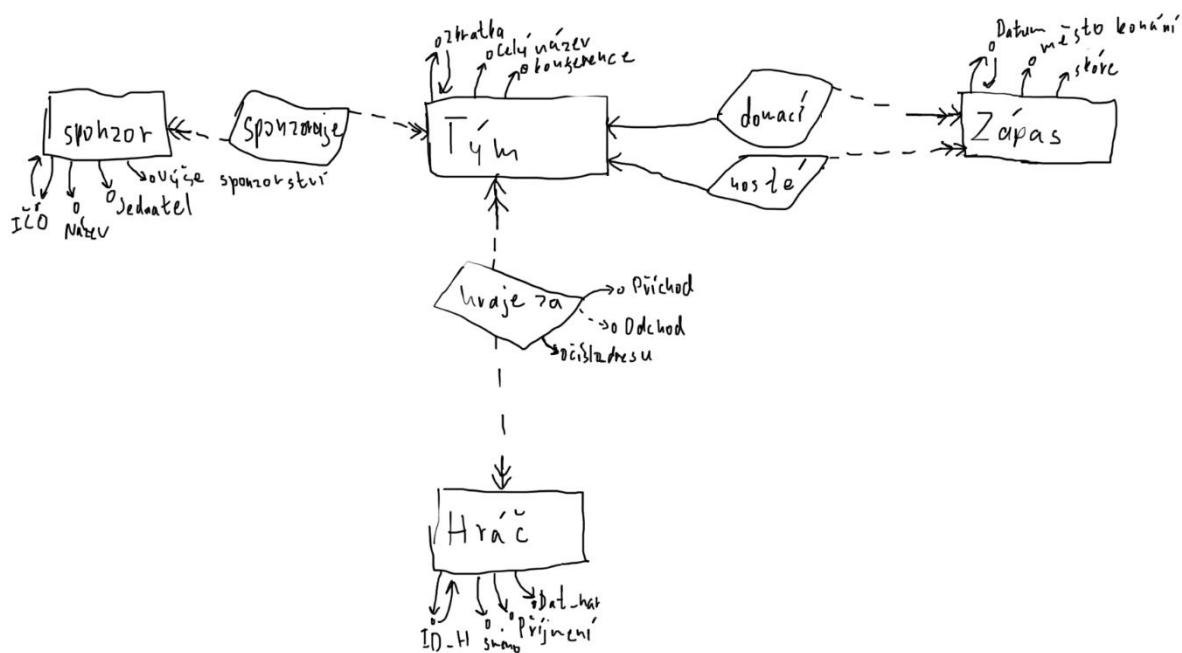
Liga potřebuje pro svou působnost evidovat cca 30 týmů. Je nutno mít informaci o jejich jednotlivých zápasech, které odehrály. U týmu je potřeba evidovat zkratku názvu (primární identifikátor), celý název týmu, konferenci (ve které hrají, W nebo E). Tým může být přiřazen k zápasu jako domácí nebo hostující tým, ne však jako oboje zároveň. Tým nemusí být přiřazen k žádnému zápasu a může být přiřazen k více. Tým může být spojen s více sponzory, ale taky s žádným. Za tým může hrát více hráčů, ale zatím taky žádný.

Zápas je identifikován částečně externími atributy 2 týmů (domácí a hosté, nesmí být stejné hodnoty v obou attributech) a dále datumem (kdy se zápas uskutečnil, v době mezi říjnem a dubnem). Dále evidujeme údaje o městě konání (název města, kde se zápas odehrál) a skóre (kterým zápas skončil). Zápasu musí být přiřazeny soupeřící týmy, jak právě jeden domácí, tak host, a každý tým může odehrát pouze jeden zápas denně.

Hráč nemusí zatím hrát za žádný tým a pokud hraje, tak pouze za jeden tým. Pomocí příchod a odchod evidujeme historii hráče v různých týmech. Vedeme u něho id_h (primární identifikátor), jméno, příjmení, datum narození, číslo dresu (pokud hraje za nějaký tým). Také se eviduje historie příchodu (kdy hráč přišel do týmu), popřípadě i odchodu (pokud již hráč odešel z týmu), oba údaje musí být v rozsahu 1. 7. xxxy až 30. 9. xxxy.

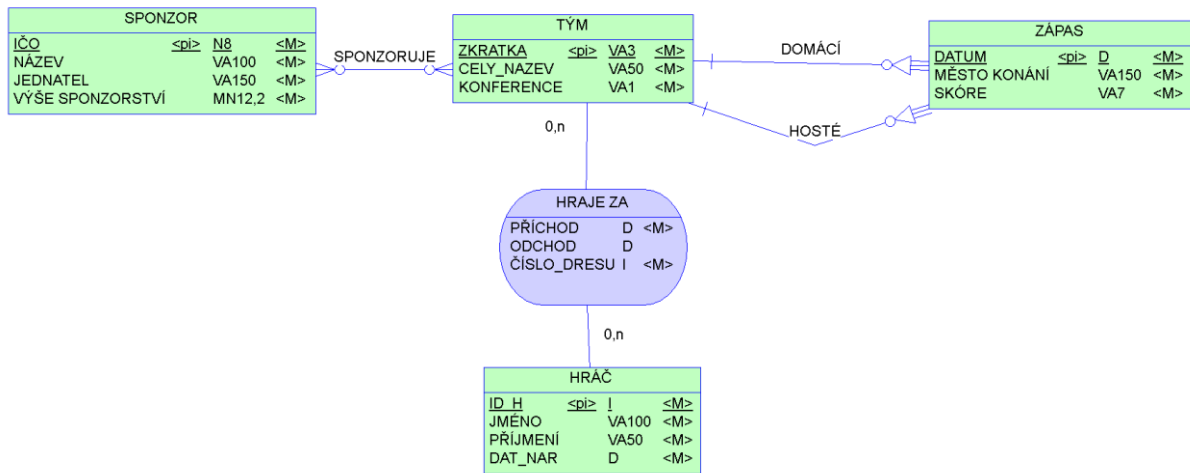
Sponzor může sponzorovat více týmů, ale taky žádný. Vedeme u něj údaje IČO (primární identifikátor), název (název právnické osoby), jméno jednatele sponzora (jméno a příjmení jednatele právnické osoby) a výši sponzorství (v Kč).

2. Konceptuální schéma reality



Obrázek 1: Evidence týmů – konceptuální schéma reality, zdroj: Jakub Hošek

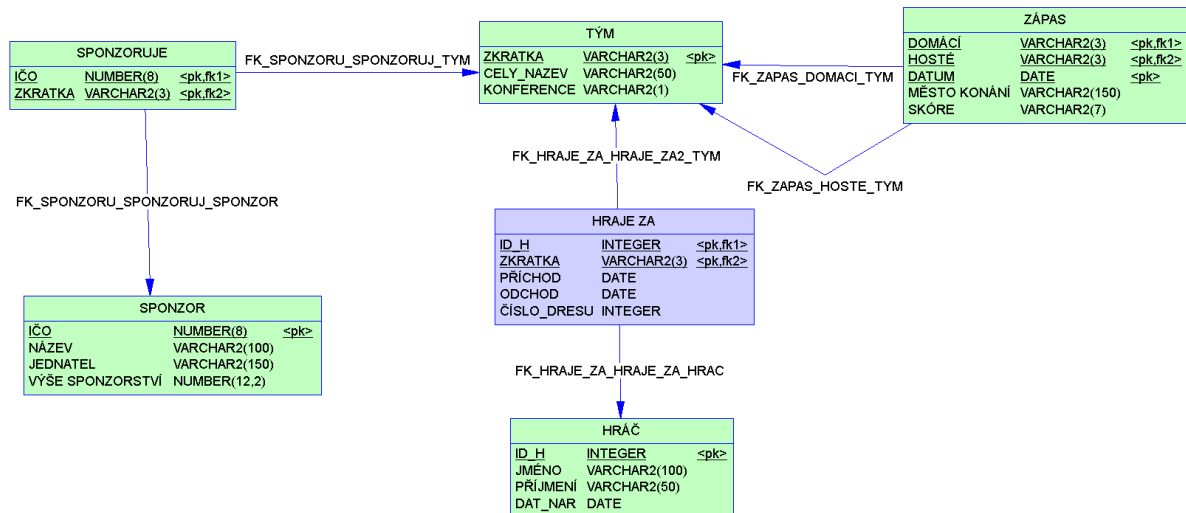
3. Konceptuální datový model



Obrázek 2: Evidence týmů – konceptuální datový model, zdroj: Jakub Hošek

4. Dokumentace databáze

4.1. Fyzický datový model



Obrázek 3: Evidence týmů – Fyzický datový model, autor: Jakub Hošek

4.2.Definice relačních tabulek a souvisejících objektů

```
/*=====*/
```

```
/* Table: HRAC */
```

```
/*=====*/
```

```
create table HRAC (
```

```
    ID_H          INTEGER          not null,
```

```
    JMENO         VARCHAR2(100)    not null,
```

```
    PRIJMENI      VARCHAR2(50)     not null,
```

```
    DAT_NAR       DATE             not null,
```

```
    constraint PK_HRAC primary key (ID_H)
```

```
);
```

```
GRANT SELECT ON HRAC TO STUDENT;
```

```
GRANT DELETE,INSERT,SELECT,UPDATE ON HRAC TO DB4IT218;
```

```
/*=====*/
```

```
/* Table: HRAJE_ZA */
```

```
/*=====*/
```

```
create table HRAJE_ZA (
```

```
    ID_H          INTEGER          not null,
```

```
    ZKRATKA       VARCHAR2(3)      not null,
```

```
    PRICHOD       DATE             not null
```

```
    constraint CKC_PRICHOD_HRAJE_ZA check (EXTRACT(MONTH FROM PRICHOD) > 4 AND  
    EXTRACT(MONTH FROM PRICHOD) < 10),
```

```
    ODCHOD        DATE             default NULL
```

```
    constraint CKC_ODCHOD_HRAJE_ZA check (EXTRACT(MONTH FROM ODCHOD) > 4 AND  
    EXTRACT(MONTH FROM ODCHOD) < 10),
```

```
    CISLO_DRESU   INTEGER          not null
```

```
    constraint CKC_CISLO_DRESU_HRAJE_ZA check (CISLO_DRESU between 0 and 99),
```

```
    constraint PK_HRAJE_ZA primary key (ID_H, ZKRATKA),
```

```
    constraint CKT_HRAJE_ZA check (PRICHOD <= ODCHOD)
```

```
);
```

```

/*=====*/
/* Index: HRAJE_ZA_FK                                */
/*=====*/
create index HRAJE_ZA_FK on HRAJE_ZA (
    ID_H ASC
);

```

```

/*=====*/
/* Index: HRAJE_ZA2_FK                                */
/*=====*/
create index HRAJE_ZA2_FK on HRAJE_ZA (
    ZKRATKA ASC
);

```

```

GRANT SELECT ON HRAJE_ZA TO STUDENT;
GRANT DELETE,INSERT,SELECT,UPDATE ON HRAJE_ZA TO DB4IT218;

```

```

/*=====*/
/* Table: SPONZOR                                */
/*=====*/
create table SPONZOR (
    ICO          NUMBER(8)          not null
        constraint CKC_ICO_SPONZOR check (ICO >= 0),
    NAZEV        VARCHAR2(100)      not null,
    JEDNATEL     VARCHAR2(150)      not null,
    VYSE_SPONZORSTVI  NUMBER(12,2)  not null
        constraint CKC_VYSE_SPONZORSTVI_SPONZOR check (VYSE_SPONZORSTVI >= 0),
    constraint PK_SPONZOR primary key (ICO)
);

```

```

GRANT SELECT ON SPONZOR TO STUDENT;
GRANT DELETE,INSERT,SELECT,UPDATE ON SPONZOR TO DB4IT218;

```

```

/*=====*/
/* Table: SPONZORUJE                                */
/*=====*/
create table SPONZORUJE (

```



```
ICO          NUMBER(8)          not null,  
ZKRATKA      VARCHAR2(3)        not null,  
constraint PK_SPONZORUJE primary key (ICO, ZKRATKA)  
);
```

```
/*=====*/  
/* Index: SPONZORUJE_FK          */  
/*=====*/  
create index SPONZORUJE_FK on SPONZORUJE (  
    ICO ASC  
);
```

```
/*=====*/  
/* Index: SPONZORUJE2_FK        */  
/*=====*/  
create index SPONZORUJE2_FK on SPONZORUJE (  
    ZKRATKA ASC  
);
```

```
GRANT SELECT ON SPONZORUJE TO STUDENT;
```

```
GRANT DELETE,INSERT,SELECT,UPDATE ON SPONZORUJE TO DB4IT218;
```

```

/*=====*/
/* Table: TYM */
/*=====*/
create table TYM (
    ZKRATKA          VARCHAR2(3)          not null
        constraint CKC_ZKRATKA_TYM check (LENGTH(ZKRATKA) = 3),
    CELY_NAZEV       VARCHAR2(50)         not null,
    KONFERENCE       VARCHAR2(1)          not null
        constraint CKC_KONFERENCE_TYM check (KONFERENCE in ('W','E')),
        constraint PK_TYM primary key (ZKRATKA)
);

GRANT SELECT ON TYM TO STUDENT;

GRANT DELETE,INSERT,SELECT,UPDATE ON TYM TO DB4IT218;

/*=====*/
/* Table: ZAPAS */
/*=====*/
create table ZAPAS (
    DOMACI           VARCHAR2(3)          not null,
    HOSTE            VARCHAR2(3)          not null,
    DATUM            DATE                 not null
        constraint CKC_DATUM_ZAPAS check( EXTRACT(MONTH FROM DATUM) >= 10 OR
        EXTRACT(MONTH FROM DATUM)<= 4),
    MESTO_KONANI     VARCHAR2(150)        not null,
    SKORE            VARCHAR2(7)          not null
        constraint CKC_SKORE_ZAPAS check(INSTR(SKORE, ':', 1) > 0 AND
        (SUBSTR( SKORE, 0, INSTR(SKORE, ':', 1) - 1) >= 0 AND SUBSTR( SKORE, 0, INSTR(SKORE,
        ':', 1) - 1) is not null) AND
        (SUBSTR( SKORE, INSTR(SKORE, ':', 1) + 1, LENGTH(SKORE)) >= 0 AND SUBSTR( SKORE,
        INSTR(SKORE, ':', 1) + 1, LENGTH(SKORE)) is not null) AND
        SUBSTR( SKORE, 0, INSTR(SKORE, ':', 1) - 1) != SUBSTR( SKORE, INSTR(SKORE, ':', 1) + 1,
        LENGTH(SKORE))),
        constraint PK_ZAPAS primary key (DOMACI, HOSTE, DATUM),
        constraint CKT_ZAPAS check (DOMACI != HOSTE)
);

/*=====*/

```

```

/* Index: DOMACI_FK                                */
/*=====*/
create index DOMACI_FK on ZAPAS (
    DOMACI ASC
);

/*=====*/
/* Index: HOSTE_FK                                */
/*=====*/
create index HOSTE_FK on ZAPAS (
    HOSTE ASC
);

GRANT SELECT ON ZAPAS TO STUDENT;
GRANT DELETE,INSERT,SELECT,UPDATE ON ZAPAS TO DB4IT218;

alter table HRAJE_ZA
    add constraint FK_HRAJE_ZA_HRAJE_ZA_HRAC foreign key (ID_H)
        references HRAC (ID_H)
        on delete cascade;

alter table HRAJE_ZA
    add constraint FK_HRAJE_ZA_HRAJE_ZA2_TYM foreign key (ZKRATKA)
        references TYM (ZKRATKA);

alter table SPONZORUJE
    add constraint FK_SPONZORU_SPONZORUJ_SPONZOR foreign key (ICO)
        references SPONZOR (ICO)
        on delete cascade;

alter table SPONZORUJE
    add constraint FK_SPONZORU_SPONZORUJ_TYM foreign key (ZKRATKA)
        references TYM (ZKRATKA);

```

```
alter table ZAPAS
```

```
add constraint FK_ZAPAS_DOMACI_TYM foreign key (DOMACI)  
references TYM (ZKRATKA);
```

```
alter table ZAPAS
```

```
add constraint FK_ZAPAS_HOSTE_TYM foreign key (HOSTE)  
references TYM (ZKRATKA);
```

```
create or replace trigger viceZapasuJednohoTymuVJedenDen
```

```
before insert or update on ZAPAS
```

```
for each row
```

```
DECLARE
```

```
val number:=0;
```

```
BEGIN
```

```
val := porovnani(:new.domaci, :new.hoste, :new.datum);
```

```
if(val = 1)
```

```
then
```

```
raise_application_error(-20000, 'domáci tým, již tento den zápas hrál');
```

```
end if;
```

```
if(val = 2)
```

```
then
```

```
raise_application_error(-20000, 'hostující tým, již tento den zápas hrál');
```

```
end if;
```

```
if(val = 3)
```

```
then
```

```
raise_application_error(-20000, 'oba týmy, již tento den zápas hrály');
```

```
end if;
```

```
END;
```

```
/
```

```
create or replace function porovnani(domaci_func in VARCHAR2, hoste_func in VARCHAR2,  
datum_func in DATE)
```

```
return number as val number;
```

```
domaci_pocet number := 0;
```

```
hoste_pocet number := 0;
```

```
BEGIN
```

```
val:=0;
```

```
select count(*) into domaci_pocet
```

```
from zapas
```

```
where datum_func = datum and
```

```
(domaci like domaci_func or hoste like domaci_func);
```

```
select count(*) into hoste_pocet
```

```
from zapas
```

```
where datum_func = datum and
```

```
(hoste like hoste_func or domaci like hoste_func);
```

```
if (domaci_pocet > 0) then val:= val + 1;
```

```
end if;
```

```
if (hoste_pocet > 0) then val:= val + 2;
```

```
end if;
```

```
return(val);
```

```
END;
```

```
/
```

```
create sequence hrac_seq
```

```
start with 1
```

```
increment by 1
```

```
nomaxvalue
```

```
nocycle;
```

create or replace trigger id_h

before insert on HRAC

for each row

BEGIN

select hrac_seq.nextval into :new.id_h

from dual;

END;

create or replace trigger kontrolaExistenceHrajeZa

before insert or update on HRAJE_ZA

for each row

DECLARE

prichod_trig := 0;

BEGIN

select count(*) into prichod_trig

from hraje_za

where id_h = :new.id_h and

(odchod is null or :new.prichod <= odchod);

if(prichod_trig > 0) then

raise_application_error(-20000, 'hráč již hraje za jiný tým');

end if;

END;

/

4.3.Integritní omezení

4.3.1. HRAC

Entitní integrita

Atributy tvořící primární klíč: ID_H

SQL kód pro přiřazení primárního klíče:

constraint PK_HRAC primary key (ID_H)

Doménová integrita

Sloupec ID_H je tvořen interní sekvencí

Popis: ID_H je umělým klíčem vytvořený systémem pro jednoduchou identifikaci hráče a zadání uživatele nemá vliv.

SQL kód omezení:

create or replace trigger id_h

before insert on HRAC

for each row

BEGIN

select hrac_seq.nextval into :new.id_h

from dual;

END;

Referenční integrita

V tabulce HRAC žádný ze sloupců není cizím klíčem.

4.3.2. HRAJE_ZA

Entitní integrita

Atributy tvořící primární klíč: ID_H, ZKRATKA

SQL kód pro přiřazení primárního klíče:

constraint PK_HRAJE_ZA primary key (ID_H, ZKRATKA)

Doménová integrita

Sloupec PRICHOD musí nabývat hodnoty měsíců mezi 5 až 9 (včetně obou).

Popis: Hráč může přijít pouze v mezi sezónní době, kdy se nehrají žádné zápasy.

SQL kód omezení:

constraint CKC_PRICHOD_HRAJE_ZA check (EXTRACT(MONTH FROM PRICHOD) > 4 AND

EXTRACT(MONTH FROM PRICHOD) < 10)

Sloupec ODCOD musí nabývat hodnoty měsíců mezi 5 až 9 (včetně obou).

Popis: Hráč může odejít pouze v mezi sezónní době, kdy se nehrají žádné zápasy.

SQL kód omezení:

```
constraint CKC_ODCHOD_HRAJE_ZA check (EXTRACT(MONTH FROM ODCHOD) > 4 AND  
EXTRACT(MONTH FROM ODCHOD) < 10)
```

Sloupec PRICHOD musí nabývat menší hodnoty než sloupec ODCHOD

Popis: Hráč nemůže odejít dříve než vůbec do týmu přišel, proto PRICHOD musí nabývat menší nebo stejnou hodnotu jako ODCHOD.

SQL kód omezení:

```
constraint CKT_HRAJE_ZA check (PRICHOD <= ODCHOD)
```

Referenční integrita

Sloupec ID_H v tabulce HRAJE_ZA představuje cizí klíč

Popis: Každá vazba hraje musí mít hráče, který hraje za nějaký tým. Hráč je jedním z hráčů v tabulce HRAC. Sloupec ID_H v HRAJE_ZA tedy obsahuje identifikační číslo hráče, který hraje za daný tým.

Řešení referenční integrity operace DELETE: CASCADE.

SQL kód omezení:

```
alter table HRAJE_ZA
```

```
add constraint FK_HRAJE_ZA_HRAJE_ZA_HRAC foreign key (ID_H)
```

```
references HRAC (ID_H);
```

Sloupec ZKRATKA v tabulce HRAJE_ZA představuje cizí klíč

Popis: Každá vazba hraje musí mít tým, za který hraje nějaký hráč. Tým je jedním z týmů v tabulce TYM. Sloupec ZKRATKA v HRAJE_ZA tedy obsahuje primární klíč týmu, za který hráč hraje.

Řešení referenční integrity operace DELETE: RESTRICT.

SQL kód omezení:

```
alter table HRAJE_ZA
```

```
add constraint FK_HRAJE_ZA_HRAJE_ZA2_TYM foreign key (ZKRATKA)
```

```
references TYM (ZKRATKA);
```

4.3.3. SPONZOR

Entitní integrita

Atributy tvořící primární klíč: ICO

SQL kód pro přiřazení primárního klíče:

```
constraint PK_SPONZOR primary key (ICO)
```

Doménová integrita

ICO musí nabývat nezáporných hodnot

Popis: Identifikační číslo je vždy nezáporné a je osmimístné.

SQL kód omezení:

```
constraint CKC_ICO_SPONZOR check (ICO >= 0 AND LENGTH(ICO) = 8)
```


VYSE_SPONZORSTVI nabývá nezáporných hodnot

Popis: Sponzorství je podpora určité osoby (fyzické i právnické), kde je očekáván marketingový přínos pro sponzora.

SQL kód omezení:

constraint CKC_VYSE_SPONZORSTVI_SPONZOR check (VYSE_SPONZORSTVI >= 0)

Referenční integrita

V tabulce SPONZOR žádný ze sloupců není cizím klíčem.

4.3.4. SPONZORUJE

Entitní integrita

Atributy tvořící primární klíč: ICO, ZKRATKA

SQL kód pro přiřazení primárního klíče:

constraint PK_SPONZORUJE primary key (ICO, ZKRATKA)

Doménová integrita

V tabulce SPONZORUJE není definováno žádné omezení, které by upravovalo doménovou integritu.

Referenční integrita

Sloupec ICO v tabulce SPONZORUJE představuje cizí klíč

Popis: Tabulka SPONZORUJE zajišťuje realizaci vztahu M:N mezi entitami SPONZOR a TYM. Záznamy jsou tak tvořeny pouze cizími klíči. Sloupec ICO v tabulce SPONZORUJE tedy obsahuje identifikační číslo sponzora.

Řešení referenční integrity operace DELETE: CASCADE.

SQL kód omezení:

alter table SPONZORUJE

add constraint FK_SPONZORU_SPONZORUJ_SPONZOR foreign key (ICO)

references SPONZOR (ICO)

on delete cascade;

Sloupec ZKRATKA v tabulce SPONZORUJE představuje cizí klíč

Popis: Tabulka SPONZORUJE zajišťuje realizaci vztahu M:N mezi entitami SPONZOR a TYM. Záznamy jsou tak tvořeny pouze cizími klíči. Sloupec ZKRATKA v tabulce SPONZORUJE tedy obsahuje primární klíč příslušného týmu.

Řešení referenční integrity operace DELETE: RESTRICT.

SQL kód omezení:

alter table SPONZORUJE

add constraint FK_SPONZORU_SPONZORUJ_TYM foreign key (ZKRATKA)

references TYM (ZKRATKA);

4.3.5. TYM

Entitní integrita

Atributy tvořící primární klíč: ZKRATKA

SQL kód pro přiřazení primárního klíče:

constraint PK_TYM primary key (ZKRATKA)

Doménová integrita

Sloupec ZKRATKA musí mít délku 3 znaky

Popis: Tým je identifikován zkratkou svého názvu o délce 3 znaků.

SQL kód omezení:

constraint CKC_ZKRATKA_TYM check (LENGTH(ZKRATKA) = 3)

Sloupec KONFERENCE musí nabývat hodnot 'W' nebo 'E'

Popis: Týmy jsou rozděleny do západní (W) a východní (E) konference.

SQL kód omezení:

constraint CKC_KONFERENCE_TYM check (KONFERENCE in ('W','E'))

Referenční integrita

V tabulce TYM žádný ze sloupců není cizím klíčem.

4.3.6. ZAPAS

Entitní integrita

Atributy tvořící primární klíč: DOMACI, HOSTE, DATUM

SQL kód pro přiřazení primárního klíče:

constraint PK_ZAPAS primary key (DOMACI, HOSTE, DATUM)

Doménová integrita

Sloupec DATUM musí nabývat hodnot měsíců 10 až 4 (oba včetně)

Popis:

SQL kód omezení:

constraint CKC_DATUM_ZAPAS check(EXTRACT(MONTH FROM DATUM) >= 10 OR
EXTRACT(MONTH FROM DATUM)<= 4)

Sloupec SKORE musí mít daný formát a rozdílné hodnoty (př. 9:8)

Popis: Skóre musí mít formát body domácích : body hosté, kdy body domácích a hostů se nesmí rovnat.

SQL kód omezení:

constraint CKC_SKORE_ZAPAS check(INSTR(SKORE, ':', 1) > 0 AND
(SUBSTR(SKORE, 0, INSTR(SKORE, ':', 1) - 1) >= 0 AND
SUBSTR(SKORE, 0, INSTR(SKORE, ':', 1) - 1) is not null) AND
(SUBSTR(SKORE, INSTR(SKORE, ':', 1) + 1, LENGTH(SKORE)) >= 0 AND

```
SUBSTR( SKORE, INSTR(SKORE, ':', 1) + 1, LENGTH(SKORE)) is not null) AND  
SUBSTR( SKORE, 0, INSTR(SKORE, ':', 1) - 1) != SUBSTR( SKORE, INSTR(SKORE, ':', 1) + 1,  
LENGTH(SKORE)))
```

Sloupce DOMACI a HOSTE nesmí nabývat stejné hodnoty

Popis: Nemůže hrát stejný tým jako domácí, tak jako hostující tým.

SQL kód omezení:

```
constraint CKT_ZAPAS check (DOMACI != HOSTE)
```

Jeden tým nemůže hrát více zápasů v jeden den

Popis: Při přidávání nebo upravování každého řádku se kontroluje, zda nějaký tým již v ten samý den nehrál zápas.

SQL kód omezení:

```
CREATE OR REPLACE TRIGGER viceZapasuJednohoTymuVJedenDen
```

```
BEFORE INSERT or UPDATE on ZAPAS
```

```
FOR EACH ROW
```

```
DECLARE
```

```
val number:=0;
```

```
BEGIN
```

```
    val := porovnani(:new.domaci, :new.hoste, :new.datum);
```

```
    if(val = 1)
```

```
    then
```

```
        raise_application_error(-20000, 'domácí tým, již tento den zápas hrál');
```

```
    end if;
```

```
    if(val = 2)
```

```
    then
```

```
        raise_application_error(-20000, 'hostující tým, již tento den zápas hrál');
```

```
    end if;
```

```
    if(val = 3)
```

```
    then
```

```
        raise_application_error(-20000, 'oba týmy, již tento den zápas hrály');
```

```
    end if;
```

```
END;
```

```
/
```

Referenční integrita

Sloupec HOSTE v tabulce ZAPAS představuje cizí klíč

Popis: Každá instance zápasu má jeden hostující tým. Tým je jeden z týmu v tabulce TYM. Sloupec HOSTE v ZAPAS tedy obsahuje primární klíč daného týmu (ZKRATKA).

Řešení referenční integrity operace DELETE: RESTRICT.

SQL kód omezení:

```
alter table ZAPAS  
add constraint FK_ZAPAS_HOSTE_TYM foreign key (HOSTE)  
references TYM (ZKRATKA);
```

Sloupec DOMACI v tabulce ZAPAS představuje cizí klíč

Popis: Každá instance zápasu má jeden domácí tým. Tým je jeden z týmu v tabulce TYM. Sloupec DOMACI v ZAPAS tedy obsahuje primární klíč daného týmu (ZKRATKA).

Řešení referenční integrity operace DELETE: RESTRICT.

SQL kód omezení:

```
alter table ZAPAS  
add constraint FK_ZAPAS_DOMACI_TYM foreign key (DOMACI)  
references TYM (ZKRATKA);
```

4.4. Definice přístupových práv

4.4.1. Přřazení přístupových práv pro uživatele STUDENT

```
GRANT SELECT ON HRAC TO STUDENT;  
GRANT SELECT ON HRAJE_ZA TO STUDENT;  
GRANT SELECT ON SPONZOR TO STUDENT;  
GRANT SELECT ON SPONZORUJE TO STUDENT;  
GRANT SELECT ON TYM TO STUDENT;  
GRANT SELECT ON ZAPAS TO STUDENT;
```

4.4.2. Přřazení přístupových práv pro uživatele DB4IT218

```
GRANT DELETE,INSERT,SELECT,UPDATE ON HRAC TO DB4IT218;  
GRANT DELETE,INSERT,SELECT,UPDATE ON HRAJE_ZA TO DB4IT218;  
GRANT DELETE,INSERT,SELECT,UPDATE ON SPONZOR TO DB4IT218;  
GRANT DELETE,INSERT,SELECT,UPDATE ON SPONZORUJE TO DB4IT218;  
GRANT DELETE,INSERT,SELECT,UPDATE ON TYM TO DB4IT218;  
GRANT DELETE,INSERT,SELECT,UPDATE ON ZAPAS TO DB4IT218;
```

4.5. Definice dalších databázových objektů

4.5.1. Sekvence čísel

Účel objektu: Každý hráč je identifikován přiřazeným číslem, které mu je přiřazeno během vložení. Sekvence generuje čísla, které jsou hráčům přiřazována.

SQL kód:

```
create sequence hrac_seq
```

```
start with 1
```

```
increment by 1
```

```
nomaxvalue
```

```
nocycle;
```

4.5.2. Funkce pro kontrolu zápasů

Účel objektu: Funkce, která se volá při vložení nebo úpravě zápasu. Kontroluje se zda zadané týmy již nehrály nějaký zápas v zadaný den.

SQL kód:

```
create or replace function porovnani(domaci_func in VARCHAR2, hoste_func in VARCHAR2,  
datum_func in DATE)
```

```
return number as val number;
```

```
domaci_pocet number := 0;
```

```
hoste_pocet number := 0;
```

```
BEGIN
```

```
val:=0;
```

```
select count(*) into domaci_pocet
```

```
from zapas
```

```
where datum_func = datum and
```

```
(domaci like domaci_func or hoste like domaci_func);
```

```
select count(*) into hoste_pocet
```

```
from zapas
```

```
where datum_func = datum and
```

```
(hoste like hoste_func or domaci like hoste_func);
```

```
if (domaci_pocet > 0) then val:= val + 1;
```

```
end if;
```

```
if (hoste_pocet > 0) then val:= val + 2;  
end if;
```

```
return(val);
```

```
END;
```

```
/
```

5. Obsah databáze

5.1. SQL příkazy pro naplnění databáze daty

```
INSERT INTO "HOSJ03"."HRAC" (ID_H, JMENO, PRIJMENI, DAT_NAR) VALUES ('1', 'Jan', 'Koch',  
TO_DATE('1988-10-05 00:00:00', 'YYYY-MM-DD HH24:MI:SS'))
```

```
INSERT INTO "HOSJ03"."HRAC" (JMENO, PRIJMENI, DAT_NAR) VALUES ('Drahoslav', 'Kohout',  
TO_DATE('1990-07-28 00:00:00', 'YYYY-MM-DD HH24:MI:SS'))
```

```
INSERT INTO "HOSJ03"."HRAC" (JMENO, PRIJMENI, DAT_NAR) VALUES ('Otakar', 'Vrabec',  
TO_DATE('1993-03-22 00:00:00', 'YYYY-MM-DD HH24:MI:SS'))
```

```
INSERT INTO "HOSJ03"."HRAC" (JMENO, PRIJMENI, DAT_NAR) VALUES ('Ingrid', 'Burian',  
TO_DATE('1987-06-15 00:00:00', 'YYYY-MM-DD HH24:MI:SS'))
```

```
INSERT INTO "HOSJ03"."HRAC" (JMENO, PRIJMENI, DAT_NAR) VALUES ('Valdemar', 'Pavliš',  
TO_DATE('1998-10-10 00:00:00', 'YYYY-MM-DD HH24:MI:SS'))
```

```
INSERT INTO "HOSJ03"."SPONZOR" (ICO, NAZEV, JEDNATEL, VYSE_SPONZORSTVI) VALUES  
(12345678, 'Under Armour', 'Ivan Hlavička', 132456)
```

```
INSERT INTO "HOSJ03"."SPONZOR" (ICO, NAZEV, JEDNATEL, VYSE_SPONZORSTVI) VALUES  
(23456789, 'Gatorade', 'Valentýn Mráček', 987564)
```

```
INSERT INTO "HOSJ03"."SPONZOR" (ICO, NAZEV, JEDNATEL, VYSE_SPONZORSTVI) VALUES  
(87654321, 'NETFLIX', 'Luboš Zach', 999999)
```

```
INSERT INTO "HOSJ03"."TYM" (ZKRATKA, CELY_NAZEV, KONFERENCE) VALUES ('GSW',  
'Golden State Warriors', 'W')
```

```
INSERT INTO "HOSJ03"."TYM" (ZKRATKA, CELY_NAZEV, KONFERENCE) VALUES ('LAL', 'Los  
Angles Lakers', 'W')
```

```
INSERT INTO "HOSJ03"."TYM" (ZKRATKA, CELY_NAZEV, KONFERENCE) VALUES ('BOS',  
'Boston Celtics', 'E')
```

```
INSERT INTO "HOSJ03"."TYM" (ZKRATKA, CELY_NAZEV, KONFERENCE) VALUES ('CHI',  
'Chicago Bulls', 'E')
```

```
INSERT INTO "HOSJ03"."TYM" (ZKRATKA, CELY_NAZEV, KONFERENCE) VALUES ('DEN',  
'Denver Nuggets', 'W')
```

```
INSERT INTO "HOSJ03"."SPONZORUJE" (ICO, ZKRATKA) VALUES ('12345678', 'GSW')
```

```
INSERT INTO "HOSJ03"."SPONZORUJE" (ICO, ZKRATKA) VALUES ('23456789', 'GSW')
```

```
INSERT INTO "HOSJ03"."SPONZORUJE" (ICO, ZKRATKA) VALUES ('23456789', 'CHI')
```

```
INSERT INTO "HOSJ03"."SPONZORUJE" (ICO, ZKRATKA) VALUES ('23456789', 'LAL')
```

```
INSERT INTO "HOSJ03"."HRAJE_ZA" (ID_H, ZKRATKA, PRICHOD, CISLO_DRESU) VALUES ('1',  
'GSW', TO_DATE('2020-07-10 00:00:00', 'YYYY-MM-DD HH24:MI:SS'), 35)
```

```
INSERT INTO "HOSJ03"."HRAJE_ZA" (ID_H, ZKRATKA, PRICHOD, CISLO_DRESU) VALUES ('2',  
'CHI', TO_DATE('2019-09-30 00:00:00', 'YYYY-MM-DD HH24:MI:SS'), 12)
```

```
INSERT INTO "HOSJ03"."HRAJE_ZA" (ID_H, ZKRATKA, PRICHOD, CISLO_DRESU) VALUES ('3',  
'BOS', TO_DATE('2015-08-13 00:00:00', 'YYYY-MM-DD HH24:MI:SS'), 0)
```

```
INSERT INTO "HOSJ03"."HRAJE_ZA" (ID_H, ZKRATKA, PRICHOD, CISLO_DRESU) VALUES ('5',  
'LAL', TO_DATE('2018-05-01 00:00:00', 'YYYY-MM-DD HH24:MI:SS'), 3)
```



```
INSERT INTO "HOSJ03"."ZAPAS" (DOMACI, HOSTE, DATUM, MESTO_KONANI, SKORE) VALUES  
( 'CHI', 'GSW', TO_DATE('2020-12-28 00:00:00', 'YYYY-MM-DD HH24:MI:SS'), 'Chicago', '95:103')
```

```
INSERT INTO "HOSJ03"."ZAPAS" (DOMACI, HOSTE, DATUM, MESTO_KONANI, SKORE) VALUES  
( 'BOS', 'LAL', TO_DATE('2021-01-31 00:00:00', 'YYYY-MM-DD HH24:MI:SS'), 'Boston', '84:99')
```

```
INSERT INTO "HOSJ03"."ZAPAS" (DOMACI, HOSTE, DATUM, MESTO_KONANI, SKORE) VALUES  
( 'GSW', 'LAL', TO_DATE('2021-01-05 00:00:00', 'YYYY-MM-DD HH24:MI:SS'), 'San Francisco',  
'87:89')
```

5.2.Opis vložených dat

5.2.1. Tabulka HRAC

ID_H	JMENO	PRIJMENI	DAT_NAR
1	Jan	Koch	05.10.88
2	Drahoslav	Kohout	28.07.90
3	Otakar	Vrabec	22.03.93
4	Ingrid	Burian	15.06.87
5	Valdemar	Pavliš	10.10.98

5.2.2. Tabulka HRAJE_ZA

ID_H	ZKRATKA	PRICHOD	ODCHOD	CISLO_DRESU
1	GSW	10.07.20		35
2	CHI	30.09.19		12
3	BOS	13.08.15		0
5	LAL	01.05.18		3

5.2.3. Tabulka SPONZOR

ICO	NAZEV	JEDNATEL	VYSE_SPONZORSTVI
12345678	Under Armour	Ivan Hlavička	132456
23456789	Gatorade	Valentýn Mráček	987564
87654321	NETFLIX	Luboš Zach	999999

5.2.4. Tabulka SPONZORUJE

ICO	ZKRATKA
12345678	GSW
23456789	CHI
23456789	GSW
23456789	LAL

5.2.5. Tabulka TYM

ZKRATKA	CELY_NAZEV	KONFERENCE
GSW	Golden State Warriors	W
LAL	Los Angles Lakers	W
BOS	Boston Celtics	E
CHI	Chicago Bulls	E
DEN	Denver Nuggets	W

5.2.6. Tabulka ZAPAS

DOMACI	HOSTE	DATUM	MĚSTO_KONANI	SKORE
CHI	GSW	28.12.20	Chicago	95:103
BOS	LAL	31.01.21	Boston	84:99
GSW	LAL	05.01.21	San Francisco	87:89