/\*Funkcia, ktora skontroluje datum ci je v poriadku \*/

CREATE DEFINER=`root`@`localhost` FUNCTION `getDayName`(day VARCHAR(20)) RETURNS varchar(10) CHARSET latin1

BEGIN

DECLARE dayName VARCHAR(10);

SELECT DAYNAME(str\_to\_date(day, '%Y-%m-%d')) into dayName;

IF day <= (select sysdate()) THEN

RETURN 'wrong day';

ELSEIF dayName in ('Sunday','Saturday') THEN

RETURN 'weekend';

END IF;

RETURN NULL;

END

/\*Funkcia, ktora skontroluje ci sa technik na danu sluzbu nachadza vo firme v dany datum\*/

CREATE DEFINER=`root`@`localhost` FUNCTION `get\_employee`(ico\_ char(8) ,service\_id\_ INTEGER) RETURNS tinyint(1)

BEGIN

if exists (select \* from employee where ico = ico\_ AND (termination\_date is null or termination\_date >= (select sysdate())) AND  work\_position in (select type from services where service\_id = service\_id\_)) then

return true;

else

return false;

end if;

END

/\* funkcia nam vrati absenciu zamestnance v dany datum \*/

CREATE DEFINER=`root`@`localhost` FUNCTION `get\_absence`($ico\_ CHAR(11), $service\_id\_ INT,$date\_ CHAR(10)) RETURNS tinyint(1)

BEGIN

DECLARE absence\_id\_ INT;

SELECT

   a.absence\_id

INTO absence\_id\_ FROM

   absence a

       JOIN

   employee e USING (hire\_date , ico , identification\_no)

WHERE

   e.ico = $ico\_

       AND (e.termination\_date IS NULL

       OR e.termination\_date >= (SELECT SYSDATE()))

       AND e.work\_position IN (SELECT

           type

       FROM

           services

       WHERE

           service\_id = $service\_id\_)

       AND (STR\_TO\_DATE($date\_, '%Y-%m-%d') BETWEEN a.absence\_from AND a.absence\_to);

IF (absence\_id\_ IS NOT NULL) THEN

RETURN TRUE;

ELSE

RETURN FALSE;

END IF;

END

/\* funkcia ktora zisti ci v danom case zamestnanec pracuje \*/

CREATE DEFINER=`root`@`localhost` FUNCTION `get\_employee\_work\_time`(ico\_ char(8) ,service\_id\_ INTEGER, order\_hour\_ INTEGER) RETURNS tinyint(1)

BEGIN

declare identification\_no\_ CHAR(11);

declare type\_ VARCHAR(20);

declare hour\_duration\_ INTEGER;

SELECT

   type, hour\_duration

INTO type\_ , hour\_duration\_ FROM

   services

WHERE

   service\_id = service\_id\_;

SELECT

   identification\_no

INTO identification\_no\_ FROM

   employee

WHERE

   ico = ico\_

       AND (termination\_date IS NULL

       OR termination\_date >= (SELECT SYSDATE()))

       AND work\_position = type\_

       AND order\_hour\_ >= working\_hour\_start

       AND ((order\_hour\_ + hour\_duration\_) <= working\_hour\_end);

IF (identification\_no\_ IS NOT NULL) THEN

RETURN TRUE;

ELSE

RETURN FALSE;

END IF;

END

**/\*Funkcia ktora skontroluje ci na dany den, firmu a hodinu uz nieje dana rezervacia \*/**

CREATE DEFINER=`root`@`localhost` FUNCTION `get\_reservations`(ico\_ CHAR(8), reservation\_date\_ CHAR(10), hour\_ INTEGER) RETURNS tinyint(1)

BEGIN

DECLARE repair\_time\_ TIME;

DECLARE hour\_duration\_ INTEGER;

DECLARE v\_last\_row\_fetched BOOLEAN;

DECLARE reservation\_cursor CURSOR FOR

SELECT

   TIME(r.repair\_date), s.hour\_duration

FROM

   reservation r

       JOIN

   services s USING (service\_id)

WHERE

   r.ico = ico\_

       AND DATE(r.repair\_date) = STR\_TO\_DATE(reservation\_date\_, '%Y-%m-%d');

DECLARE CONTINUE HANDLER FOR NOT FOUND SET  v\_last\_row\_fetched = 1;

OPEN reservation\_cursor;

get\_reservation: LOOP

FETCH reservation\_cursor INTO repair\_time\_,hour\_duration\_;

   IF (v\_last\_row\_fetched = 1) THEN

           LEAVE get\_reservation;

END IF;

IF( (TIME\_TO\_SEC(TIME(repair\_time\_)) <= (hour\_\*3600)   && (hour\_\*3600) < (TIME\_TO\_SEC(TIME(repair\_time\_)) + (hour\_duration\_\*3600) ))) THEN

RETURN TRUE;

END IF;

END LOOP get\_reservation;

CLOSE reservation\_cursor;

SET  v\_last\_row\_fetched = 0;

RETURN FALSE;

END

// Funkcia skontroluje uzivatelov vstup ci zadal spravne mesto psc a krajinu

CREATE DEFINER=`root`@`localhost` FUNCTION `check\_town`(town\_id\_ CHAR(5), town\_name\_ VARCHAR(100), country\_id\_ CHAR(3)) RETURNS tinyint(1)

BEGIN

DECLARE var\_town\_name\_ VARCHAR(100);

DECLARE var\_country\_id\_ CHAR(3);

SELECT

   town\_name, country\_id

INTO var\_town\_name\_ , var\_country\_id\_ FROM

   town

WHERE

   town\_id = town\_id\_;

IF EXISTS (SELECT \* FROM town WHERE town\_name = town\_name\_ AND country\_id = country\_id\_ AND town\_id != town\_id\_) THEN

RETURN 0;

END IF;

IF (var\_town\_name\_ IS NOT NULL && var\_country\_id\_ IS NOT NULL) THEN

IF (var\_town\_name\_ != town\_name\_ || var\_country\_id\_ != country\_id\_) THEN

RETURN 0;

END IF;

END IF;

RETURN 1;

END

/\*Funkcia, ktorá skontroluje rezerváciu, či je aktuálna \*/

CREATE DEFINER=`root`@`localhost` FUNCTION `check\_reservation\_date`(reservation\_id\_ INTEGER) RETURNS tinyint(1)

BEGIN

DECLARE repair\_date\_ DATE;

SELECT

   repair\_date

INTO repair\_date\_ FROM

   reservation

WHERE

   reservation\_id = reservation\_id\_;

IF(repair\_date\_ >= NOW() + INTERVAL 1 DAY) THEN

RETURN TRUE;

ELSE

RETURN FALSE;

END IF;

END

/\* Funkcia overí duplicitu pri vytváraní zákazníka \*/

CREATE DEFINER=`root`@`localhost` FUNCTION `duplicate\_customer`(identification\_no\_ CHAR(11)) RETURNS tinyint(1)

BEGIN

IF EXISTS(select \* from customer where identification\_no = identification\_no\_) THEN

RETURN TRUE;

ELSE

RETURN FALSE;

END IF;

END

/\* Funkcia, ktora zisti ci dana car brand uz nieje v systeme \*/

CREATE DEFINER=`root`@`localhost` FUNCTION `check\_car\_brand`(brand\_name\_ VARCHAR(30)) RETURNS tinyint(1)

BEGIN

IF EXISTS (SELECT \* FROM car\_brand WHERE brand\_name = brand\_name\_) THEN

RETURN TRUE;

ELSE

RETURN FALSE;

END IF;

END

/\*Funkcia zisti ci dany typ auta sa uz v systeme nenachadza \*/

CREATE DEFINER=`root`@`localhost` FUNCTION `check\_car\_type`(brand\_id\_ INTEGER, car\_type\_name\_ VARCHAR(100)) RETURNS tinyint(1)

BEGIN

IF EXISTS(SELECT \* FROM car\_type WHERE brand\_id = brand\_id\_ AND car\_type\_name = car\_type\_name\_) THEN

RETURN TRUE;

ELSE

RETURN FALSE;

END IF;

END

/\*Funkcia zisti ci dana autosciastka uz v systeme nenachadza \*/

CREATE DEFINER=`root`@`localhost` FUNCTION `check\_car\_part`(car\_type\_id\_ INTEGER, part\_name\_ VARCHAR(100)) RETURNS tinyint(1)

BEGIN

IF EXISTS(SELECT \* FROM car\_parts WHERE car\_type\_id = car\_type\_id\_ AND part\_name = part\_name\_) THEN

RETURN TRUE;

ELSE

RETURN FALSE;

END IF;

END