Zad 1

delimiter //

create trigger kreatura\_before\_insert

before insert on kreatura

for each row

begin

if new.waga<0

then

set new.waga=0;

end if;

end

//delimiter ;

Zad 2

delimiter //

create trigger wyprawa\_after\_delete

AFTER DELETE ON wyprawa

FOR EACH ROW

BEGIN

DECLARE kierownik\_nazwa VARCHAR(100);

SET kierownik\_nazwa = (SELECT kreatura.nazwa FROM kreatura WHERE idKreatury = OLD.kierownik);

INSERT INTO archiwum\_wypraw VALUES (OLD.id\_wyprawy, OLD.nazwa, OLD.data\_rozpoczecia, OLD.data\_zakonczenia, kierownik\_nazwa);

END

//delimiter ;

Zad 3

a)delimiter //

CREATE PROCEDURE eliksir\_sily(IN id int)

BEGIN

Update kreatura set udzwig = 1.2 \* udzwig where idKreatury= id;

END

//delimiter ;

b)delimiter //

CREATE FUNCTION wielkie\_litery(text varchar(255))

RETURNS VARCHAR(255)

BEGIN

DECLARE ucase VARCHAR(255);

SET @ucase = UCASE(text);

RETURN @ucase;

END

//delimiter ;

Zad 4

delimiter $$

create trigger uczestnicy\_after\_insert

after insert ON uczestnicy

for each row

begin

if ( select id\_uczestnika=14 from uczestnicy where id\_wyprawy=NEW.id\_wyprawy) AND

(select sektor=7 FROM etapy\_wyprawy where idWyprawy=NEW.id\_wyprawy)

then

insert into system\_alarmowy VALUES(default,'Tesciowa nadchodzi');

end if;

end

$$

delimiter ;

Zad 5

delimiter //

create procedure parametry\_kreatur(out parametry int)

begin

select concat("Średnia: ", avg(udzwig)," ","Suma: " sum(udzwig)," ","Maks: ", max(udzwig) into parametry from kreatura;

end

//delimiter ;