

## PROCEDURA – REALIZUJ ZAMÓWIENIE

1. SPRAWDZA CZY WIELKOŚCI KOLEKCJI ODPOWIADAJĄCYCH ZA IŁOSC ZABAWEK ORAZ LICZBE IŁOSCI ZAMÓWIONYCH ZABAWEKĄ RÓWNE
  - A. JEŚLI NIE – STOSOWNY KOMUNIKAT, STOP
  - B. JEŚLI TAK – KONTYNUUJ
2. WSTAWIA NOWY WIERSZ DO TABELI ZAMOWIENIA
3. SPRAWDZA CZY JEST DOSTĘPNA WYSTARCZAJĄCA IŁOSC ZABAWEK NA MAGAZYNIE
  - A. JEŚLI NIE – STOSOWNY KOMUNIKAT, STOP
  - B. JEŚLI TAK – KONTYNUUJ
4. ZLICZA IŁOŚĆ ZABAWEK ZAKUPIONYCH PRZEZ KLIENTA
  - A. SPRAWDZA CZY IŁOŚĆ TA JEST WIĘKSZA NIŻ 100
    - a. Jeśli tak – zmienia klienta na stałego
    - b. Jeśli nie – kontynuuj
5. OBSŁUGA WYJĄTKÓW

```
CREATE TYPE ZabawkiCollection AS TABLE OF INTEGER;
CREATE TYPE IlosciZabawekCollection AS TABLE OF INTEGER;
CREATE OR REPLACE PROCEDURE RealizujZamowienie(
    p_id_klienta INTEGER,
    p_data_zamowienia DATE,
    p_id_sposobu_zaplaty INTEGER,
    p_produkty_tab IN ZabawkiCollection,
    p_ilosci_tab IN IlosciZabawekCollection
) AS
    v_id_zamowienia INTEGER;
    v_id_pozycji_zamowienia INTEGER;
    v_liczba_zamowionych_zabawek INTEGER := 0;

BEGIN
    -- Sprawdź, czy ilości i produkty mają taką samą długość
    IF p_produkty_tab.COUNT <> p_ilosci_tab.COUNT THEN
        DBMS_OUTPUT.PUT_LINE('Błąd: Kolekcje p_produkty_tab i p_ilosci_tab muszą mieć tę samą
długość.');
```

```
        RETURN;
    END IF;

    -- Znajdź pierwszy dostępny identyfikator zamówienia
    SELECT COALESCE(MAX(id_zamowienia) + 1, 1)
    INTO v_id_zamowienia
    FROM Zamowienia;

    -- Znajdź pierwszy dostępny identyfikator pozycji zamówienia
    SELECT COALESCE(MAX(id_pozycji_zamowienia) + 1, 1)
    INTO v_id_pozycji_zamowienia
    FROM Pozycje_Zamowienia;

    -- Rozpocznij transakcję
    BEGIN
        -- Wstaw dane zamówienia do tabeli Zamowienia
        INSERT INTO Zamowienia (id_zamowienia, id_klienta, data_zamowienia, status_zamowienia,
id_sposobu_zaplaty)
        VALUES (v_id_zamowienia, p_id_klienta, p_data_zamowienia, 'W trakcie',
p_id_sposobu_zaplaty);

        -- Wstaw produkty do tabeli Pozycje_Zamowienia
        FOR i IN 1..p_produkty_tab.COUNT LOOP
            -- Sprawdź dostępność zabawek
```

```

DECLARE
    v_dostepna_ilosc INTEGER;
BEGIN
    SELECT dostepna_ilosc
    INTO v_dostepna_ilosc
    FROM Zabawki
    WHERE id_zabawki = p_produkty_tab(i);

    IF v_dostepna_ilosc < p_ilosci_tab(i) THEN
        RAISE_APPLICATION_ERROR(-20001, 'Błąd: Brak wystarczającej ilości zabawek o ID ' ||
p_produkty_tab(i));
    END IF;

    -- Odejmij zamówione ilości od dostępnych zabawek
    UPDATE Zabawki
    SET dostepna_ilosc = dostepna_ilosc - p_ilosci_tab(i)
    WHERE id_zabawki = p_produkty_tab(i);

    -- Zlicz zamówione ilości
    v_liczba_zamowionych_zabawek := v_liczba_zamowionych_zabawek + p_ilosci_tab(i);
EXCEPTION
    WHEN NO_DATA_FOUND THEN
        RAISE_APPLICATION_ERROR(-20002, 'Błąd: Zabawka o ID ' || p_produkty_tab(i) || ' nie
istnieje.');
```

```

    END;

    -- Wstaw pozycję zamówienia
    INSERT INTO Pozycje_Zamowienia (id_pozycji_zamowienia, id_zamowienia, id_zabawki,
ilosc_sztuk)
    VALUES (v_id_pozycji_zamowienia, v_id_zamowienia, p_produkty_tab(i), p_ilosci_tab(i));

    v_id_pozycji_zamowienia := v_id_pozycji_zamowienia + 1;
END LOOP;

-- Zakończ transakcję
COMMIT;

-- Sprawdzenie, czy liczba zamówionych zabawek przekracza 100
IF v_liczba_zamowionych_zabawek > 100 THEN
    -- Zmiana klienta na stałego
    UPDATE Klienci
    SET staly_klient = 1
    WHERE id_klienta = p_id_klienta;

    DBMS_OUTPUT.PUT_LINE('Klient o ID ' || p_id_klienta || ' został zmieniony na stałego
klienta.');
```

```

    END IF;

    DBMS_OUTPUT.PUT_LINE('Zamówienie o ID ' || v_id_zamowienia || ' zostało zrealizowane.');
```

```

EXCEPTION
    WHEN OTHERS THEN
        -- W przypadku błędu, wykonaj rollback
        ROLLBACK;
        DBMS_OUTPUT.PUT_LINE('Wystąpił błąd podczas realizacji zamówienia: ' || SQLERRM);
    END;
END RealizujZamowienie;
/
```

## BŁĘDY

### - BŁĄD PRZY RÓŻNEJ DŁUGOŚCI TABLIC

```
488 DECLARE
489     produkty_tab ZabawkiCollection := ZabawkiCollection(23, 2, 3);
490     ilosci_tab IlosciZabawekCollection := IlosciZabawekCollection(4, 1, 1, 3);
491 BEGIN
492     RealizujZamowienie(2, SYSDATE, 1, produkty_tab, ilosci_tab);
493 END;
```

Statement processed.

Błąd: Kolekcje p\_produkty\_tab i p\_ilosci\_tab muszą mieć tę samą długość.

### -BŁĄD - ZA MAŁO ZABAWEK W BAZIE

```
497 select * from zabawki where dostepna_ilosc < 20;
```

ID_ZABAWKI	NAZWA	CENA	DOSTEPNA_ILOSC	ID_KATEGORII	ID_MATERIALU
6	Laptop edukacyjny	89.99	15	1	11

```
487 DECLARE
488     produkty_tab ZabawkiCollection := ZabawkiCollection(6, 2, 3);
489     ilosci_tab IlosciZabawekCollection := IlosciZabawekCollection(16, 1, 1);
490 BEGIN
491     RealizujZamowienie(2, SYSDATE, 1, produkty_tab, ilosci_tab);
492 END;
```

Statement processed.

Wystąpił błąd podczas realizacji zamówienia: ORA-20001: Błąd: Brak wystarczającej ilości zabawek o ID 6

## POPRAWNE WYKONANIE

Przed wykonaniem procedury:

ID_KLIENTA	NAZWA_FIRMY	IMIE	NAZWISKO	STALY_KLIENT	ILOSC_ZAKUPIONYCH_ZABAWEK
2	Kraina Zabaw	Piotr	Kowalski	0	1

496	select * from zabawki where id_zabawki IN (23,4,5);				
497					
498					
ID_ZABAWKI	NAZWA	CENA	DOSTEPNA_ILOSC	ID_KATEGORII	ID_MATERIALU
4	Klocki drewniane	29.99	40	9	2
5	Gra planszowa "Zlap Muche"	49.99	25	7	3
23	Ksiazka Zabawkowa dla Malucha	9.99	120	2	6

Wykonanie procedury:

487	DECLARE
488	produkty_tab ZabawkiCollection := ZabawkiCollection(23, 4, 5);
489	ilosci_tab IlosciZabawekCollection := IlosciZabawekCollection(100, 15, 14);
490	BEGIN
491	RealizujZamowienie(2, SYSDATE, 3, produkty_tab, ilosci_tab);
492	END;
493	/
494	
Statement processed.	
Klient o ID 2 został zmieniony na stałego klienta.	
Zamówienie o ID 31 zostało zrealizowane.	

Efekty działania procedury:

ID_KLIENTA	NAZWA_FIRMY	IMIE	NAZWISKO	STALY_KLIENT	ILOSC_ZAKUPIONYCH_ZABAWEK
2	Kraina Zabaw	Piotr	Kowalski	1	131

496	select * from zamowienia;			
497				
498				
	ID_ZAMOWIENIA	DATA_ZAMOWIENIA	STATUS_ZAMOWIENIA	ILOSC_ZAKUPIONYCH_ZABAWEK
29	01-DEC-23	Zrealizowane	4	4
30	06-DEC-23	W trakcie	5	5
31	12-DEC-23	W trakcie	2	3

496	select * from pozycje_zamowienia;		
497			
498			
	ID_ZAMOWIENIA	ID_ZABAWKI	ILOSC_ZAKUPIONYCH_ZABAWEK
31	31	23	100
32	31	4	15
33	31	5	14

- 
1. SPRAWDZA JEŚLI STATUS JEST POPRAWNY ('W TRAKCIE', 'ZREALIZOWANE', 'ANULOWANE').
    - A. JEŚLI NIE - STOSOWNY KOMUNIKAT, STOP
    - B. JEŚLI TAK - KONTYNUUJ
  2. SPRAWDZA CZY PODANY NOWY STATUS RÓŻNI SIĘ OD OBECNEGO
    - A. JEŚLI NIE - STOSOWNY KOMUNIKAT, STOP
    - B. JEŚLI TAK - KONTYNUUJ
  3. JEŚLI ZMIENIAMY Z 'ANULOWANE' NA ('W TRAKCIE', 'ZREALIZOWANE') TO:
    - A. SPRAWDZAMY CZY JEST WYSTARCZAJĄCA LICZBA ZABAWEK W BAZIE, ABY ZREALIZOWAC ZAMOWIENIE
      - a. Jeśli tak - zmniejszamy liczbę dostępnych zabawek
      - b. Jeśli nie - stosowny komunikat
    - B. SPRAWDZAMY CZY ILOSC ZAKUPIONYCH ZABAWEK PRZEZ KLIENTA PRZEKROCZYŁA 100
      - a. Jeśli tak - zmieniamy status klienta na stałego
      - b. Jeśli nie - kontynuuj
  4. JEŚLI ZMIENIAMY Z ('W TRAKCIE', 'ZREALIZOWANE') NA ANULOWANE TO:
    - a. Dodajemy liczbę dostępnych zabawek
    - A. SPRAWDZAMY CZY ILOSC ZAKUPIONYCH ZABAWEK PRZEZ KLIENTA NIE SPADŁA PONIŻEJ 100
      - a. Jeśli tak - zmieniamy status klienta na niestałego
      - b. Jeśli nie - kontynuuj
  5. OBSŁUGA WYJĄTKÓW

```
CREATE OR REPLACE PROCEDURE ZmienStatusZamowienia(
    p_id_zamowienia INTEGER,
    p_nowy_status VARCHAR2
) AS
    v_obecny_status VARCHAR2(20);

-- Definicja kursora
CURSOR c_zabawki IS
    SELECT PZ.id_zabawki, PZ.ilosc_sztuk, Z.id_klienta
    FROM Pozycje_Zamowienia PZ
        JOIN Zamowienia Z ON PZ.id_zamowienia = Z.id_zamowienia
    WHERE PZ.id_zamowienia = p_id_zamowienia;

-- Zmienne do przechowywania danych z kursora
v_zabawka_id INTEGER;
v_ilosc_sztuk INTEGER;
v_id_klienta INTEGER;
v_dostepna_ilosc INTEGER;
v_ilosc_zamowiona_przez_klienta INTEGER;
BEGIN
    -- Sprawdzenie, czy zamówienie istnieje
    SELECT status_zamowienia
    INTO v_obecny_status
    FROM Zamowienia
    WHERE id_zamowienia = p_id_zamowienia;

    -- Sprawdzenie, czy podany status jest poprawny
    IF p_nowy_status NOT IN ('W trakcie', 'Zrealizowane', 'Anulowane') THEN
        DBMS_OUTPUT.PUT_LINE('Błąd: Niepoprawny status zamówienia.');
```

```

END IF;

-- Sprawdzenie, czy nowy status różni się od obecnego
IF p_nowy_status = v_obecny_status THEN
    DBMS_OUTPUT.PUT_LINE('Błąd: Nowy status jest taki sam jak obecny.');
```

RETURN;

```

END IF;

-- Zmiana statusu zamówienia
UPDATE Zamowienia
SET status_zamowienia = p_nowy_status
WHERE id_zamowienia = p_id_zamowienia;

-- Otwarcie kursora
OPEN c_zabawki;

-- Pętla FETCH dla kursora
LOOP
    FETCH c_zabawki INTO v_zabawka_id, v_ilosc_sztuk, v_id_klienta;
    EXIT WHEN c_zabawki%NOTFOUND;

    SELECT dostepna_ilosc
    INTO v_dostepna_ilosc
    FROM Zabawki
    WHERE id_zabawki = v_zabawka_id;

    SELECT SUM(PZ.ilosc_sztuk)
    INTO v_ilosc_zamowiona_przez_klienta
    FROM Pozycje_Zamowienia PZ
    JOIN Zamowienia Z ON PZ.id_zamowienia = Z.id_zamowienia
    WHERE Z.id_klienta = v_id_klienta;

    -- Usunięcie zabawek przypadku zmiany z na "Zrealizowane" lub "W trakcie"
    IF v_obecny_status = 'Anulowane' AND (p_nowy_status = 'W trakcie' OR p_nowy_status
= 'Zrealizowane') THEN
        IF(v_dostepna_ilosc - v_ilosc_sztuk < 0) THEN
            DBMS_OUTPUT.PUT_LINE('Niewystarczająca ilość zabawek na przeprowadzenie takiej
operacji.');
```

RETURN;

```

        END IF;
        UPDATE Zabawki
        SET dostepna_ilosc = dostepna_ilosc - v_ilosc_sztuk
        WHERE id_zabawki = v_zabawka_id;

-- Sprawdzenie czy ilość kupionych zabawek przez klienta przekroczyła 100
IF ((v_ilosc_zamowiona_przez_klienta + v_ilosc_sztuk) > 100) THEN
    -- Zmiana klienta na stałego
    UPDATE Klienci
    SET staly_klient = 1
    WHERE id_klienta = v_id_klienta;

    DBMS_OUTPUT.PUT_LINE('Zmieniono klienta o ID ' || v_id_klienta || ' na
klienta stalego.');
```

END IF;

```

        DBMS_OUTPUT.PUT_LINE('Zamówienie o ID ' || p_id_zamowienia || ' zostało
zrealizowane, zabawki usunięte z magazynu.');
```

END IF;

```

    -- Przywrocenie zabawek do magazynu w przypadku zmiany na "Anulowane"
    IF (v_obecny_status = 'Zrealizowane' OR v_obecny_status = 'W trakcie') AND
p_nowy_status = 'Anulowane' THEN
        UPDATE Zabawki
        SET dostepna_ilosc = dostepna_ilosc + v_ilosc_sztuk
        WHERE id_zabawki = v_zabawka_id;
```

-- Sprawdzenie czy ilość kupionych zabawek przez klienta nie spadła poniżej 100

```

    IF ((v_ilosc_zamowiona_przez_klienta - v_ilosc_sztuk) < 100) THEN
        -- Zmiana klienta na niestalego
        UPDATE Klienci
        SET staly_klient = 0
        WHERE id_klienta = v_id_klienta;
```

DBMS\_OUTPUT.PUT\_LINE('Zmieniono klienta o ID ' || v\_id\_klienta || ' na klienta niestalego.');

END IF;

```

        DBMS_OUTPUT.PUT_LINE('Zamówienie o ID ' || p_id_zamowienia || ' zostało
anulowane, zabawki przywrócone do magazynu.');
```

END IF;

END LOOP;

CLOSE c\_zabawki;

DBMS\_OUTPUT.PUT\_LINE('Status zamówienia o ID ' || p\_id\_zamowienia || ' został zmieniony na: ' || p\_nowy\_status);

COMMIT;

EXCEPTION

WHEN NO\_DATA\_FOUND THEN

```

    DBMS_OUTPUT.PUT_LINE('Błąd: Zamówienie o ID ' || p_id_zamowienia || ' nie
istnieje.');
```

WHEN OTHERS THEN

```

    ROLLBACK;
    DBMS_OUTPUT.PUT_LINE('Błąd: ' || SQLERRM);
```

END ZmienStatusZamowienia;

DECLARE

```

    v_id_zamowienia INTEGER := 1;
    v_nowy_status VARCHAR2(20) := 'Zrealizowane';
```

BEGIN

```

    ZmienStatusZamowienia(v_id_zamowienia, v_nowy_status);
```

END;

/

PRZYKŁADY DZIAŁANIA PROCEDURY

POPRAWNE DZIAŁANIE

ID_KLIENTA	NAZWA_FIRMY	IMIE	NAZWISKO	STALY_KLIENT
1	Magiczne Zabawki	Anna	Nowak	0

520 `select * from zamowienia;`

ID_ZAMOWIENIA	DATA_ZAMOWIENIA	STATUS_ZAMOWIENIA	ID_KLIENTA	ID_SPOSOBU_ZAPLATY
1	21-SEP-23	Anulowane	1	1

150 `DECLARE`

151 `v_id_zamowienia INTEGER := 1;`

152 `v_nowy_status VARCHAR2(20) := 'Zrealizowane';`

153 `BEGIN`

154 `ZmienStatusZamowienia(v_id_zamowienia, v_nowy_status);`

155 `END;`

156 `/`

157

Statement processed.  
Zmieniono klienta o ID 1 na klienta stalego.  
Zamówienie o ID 1 zostało zrealizowane, zabawki usunięte z magazynu.  
Zamówienie o ID 1 zostało zrealizowane, zabawki usunięte z magazynu.  
Zamówienie o ID 1 zostało zrealizowane, zabawki usunięte z magazynu.  
Status zamówienia o ID 1 został zmieniony na: Zrealizowane

ID_KLIENTA	NAZWA_FIRMY	IMIE	NAZWISKO	STALY_KLIENT
1	Magiczne Zabawki	Anna	Nowak	1



## BŁĘDY

- Użycie tego samego statusu

```
614 select * from zamowienia;
```

ID_ZAMOWIENIA	DATA_ZAMOWIENIA	STATUS_ZAMOWIENIA	ID_KLIENTA	ID_SPOSOBU_ZAPLATY
1	21-SEP-23	Anulowane	1	1
2	01-FEB-23	Zrealizowane	2	2
3	01-NOV-23	W trakcie	3	3

```
634 v DECLARE
635     v_id_zamowienia INTEGER := 2;
636     v_nowy_status VARCHAR2(20) := 'Zrealizowane';
637 v BEGIN
638     ZmienStatusZamowienia(v_id_zamowienia, v_nowy_status);
639 END;
640 /
641
```

Statement processed.  
Błąd: Nowy status jest taki sam jak obecny.

- Użycie błędnego statusu

```
634 v DECLARE
635     v_id_zamowienia INTEGER := 2;
636     v_nowy_status VARCHAR2(20) := 'Test';
637 v BEGIN
638     ZmienStatusZamowienia(v_id_zamowienia, v_nowy_status);
639 END;
640 /
641
```

Statement processed.  
Błąd: Niepoprawny status zamówienia.

- Niewystarczająca liczba zabawek

```
515 select * from zabawki;
```

ID_ZABAWKI	NAZWA	CENA	DOSTEPNA_ILOSC	ID_KATEGORII	ID_MATERIALU
1	Puzzle Edukacyjne	19.99	50	1	5

```
513 select * from pozycje_zamowienia;
```

```
514
```

ID_POZYCJI_ZAMOWIENIA	ID_ZAMOWIENIA	ID_ZABAWKI	ILOSC_SZTUK
1	1	1	95
2	1	3	2
3	1	2	1

```
550 v DECLARE
```

```
551     v_id_zamowienia INTEGER := 1;
```

```
552     v_nowy_status VARCHAR2(20) := 'Zrealizowane';
```

```
553 v BEGIN
```

```
554     ZmienStatusZamowienia(v_id_zamowienia, v_nowy_status);
```

```
555 END;
```

```
556 /
```

```
557
```

Statement processed.

Niewystarczająca ilość zabawek na przeprowadzenie takiej operacji.