

Lab2-T2

(Calculo del Precio Promedio Ponderado)

Leonardo H. Añez Vladimirovna*

*Universidad Autónoma Gabriel René Moreno,
Facultad de Ingeniería en Ciencias de la Computación y Telecomunicaciones,
Santa Cruz de la Sierra, Bolivia*

18 de octubre de 2019

El precio medio ponderado (PMP) es un método de valoración en contabilidad que se obtiene al realizar un cálculo del valor medio de las existencias que había al inicio y de las entradas ponderadas según sus cantidades.

Fórmula del precio promedio ponderado (PPP)

Para calcular el Precio Promedio Ponderado (PPP) es necesario conocer esta sencilla formula. Explicando los términos que la componen, tenemos:

- $\sum P_i$ = Suma del precio de cada bien.
- Q_i = Cantidad de bienes comprados
- $\sum Q_i$ = Cantidad total de bienes

$$PPP = \frac{\sum P_i \cdot Q_i}{\sum Q_i}$$

¿Qué supone el uso del PPP?

Por las propias condiciones propias del modelo, la utilización del precio promedio ponderado supone un continuo control de las existencias para su correcto funcionamiento, debido a que es necesario tener los datos sobre las cantidades almacenadas y las diferentes entradas y salidas de forma totalmente controlada.

Además, este criterio de valoración de existencias es bastante confiable en periodos de tiempo de estabilidad en los precios, ya que reduce los efectos producidos por las posibles oscilaciones de estos.

* Correo Electrónico: toborochi98@outlook.com

Implementacion

```
01 USE preventas;
02 /*
03      Funcion que calcula el PPP(Precio Promedio Ponderado)
04 */
05 GO
06 CREATE FUNCTION PA_PPP(@cprd INT, @calm INT)
07 RETURNS DECIMAL(12,2)
08 AS
09 BEGIN
10     DECLARE @suma_total DECIMAL(12,2)
11     DECLARE @cant_total DECIMAL(12,2)
12     DECLARE @cant DECIMAL(12,2)
13     DECLARE @prec DECIMAL(12,2)
14     DECLARE @prec_final DECIMAL(12,2)
15     SET @suma_total = 0
16     SET @cant_total = 0
17
18     DECLARE c_cursor CURSOR FOR SELECT sumi.cant, sumi.prec
19     FROM sumi
20     WHERE sumi.calm = @calm AND sumi.cprd = @cprd
21
22     OPEN c_cursor
23     FETCH c_cursor INTO @cant, @prec
24     WHILE @@FETCH_STATUS = 0
25     BEGIN
26         SET @suma_total = @suma_total + (@cant * @prec)
27         SET @cant_total = @cant_total + @cant
28         FETCH c_cursor INTO @cant, @prec
29     END
30
31     CLOSE c_cursor
32     DEALLOCATE c_cursor
33
34     SET @prec_final = @suma_total / @cant_total
35
36     RETURN @prec_final
37 END
38 GO
```