

---

## Mètode per a l'eliminació d'usuari a gestor de carteres cripto.X34597\_ca

---

Implementa el mètode `deleteUser` a la classe `CryptoApp`. Aquest mètode ha de gestionar l'eliminació completa d'un usuari del sistema, assegurant que no quedin dades inconsistents ("orfes"). Utilitzeu l'arxiu `CryptoApp.cpp` i implementeu el mètode `void CryptoApp::deleteUser(const string& name)` situat al final de l'arxiu. Haureu de:

1. Descarregar els arxius públics.
2. Canviar el nom de `CryptoApp_old.cpp` a `CryptoApp.cpp`.
3. Implementar el codi del mètode `deleteUser`:

```
// PRE: name és el nom d'usuari a eliminar
// POST: Si l'usuari existeix, elimina l'usuari i tots els seus
//       holdings associats de manera eficient. Si no, no fa res.
void CryptoApp::deleteUser(const string& name)
{
}
```

4. Comprimir en format `.tar` l'arxiu `CryptoApp.cpp`.

```
tar -cvf CryptoApp.tar CryptoApp.cpp
```

5. Pujar l'arxiu `.tar` a Jutge amb la funció implementada.

### ESPECIFICACIONS:

1. **Entrada:** Rep el nom de l'usuari (`string`) (ex: `jaume`).
2. **Context:** La classe disposa de mètodes interns que pot consultar. Per exemple, per buscar si un usuari existeix i la seva posició en `vec_users`, pot utilitzar:

```
int findUserIndexByName(string name)
```

3. **Validació:** Si l'usuari no existeix, la funció no ha d'eliminar res.
4. **Esborrat en Cascada:** Abans d'eliminar l'usuari, ha d'eliminar totes les tinences (*Holdings*) vinculades a aquest usuari del vector `vec_holdings`.
5. **Esborrat de l'Usuari:** Finalment, elimini l'usuari del vector `vec_users`.

### Observacions:

- Els vectors `vec_users` i `vec_holdings` es relacionen mitjançant l'`userID (int)`, no pel nom.
- Ha d'obtenir primer l'`ID` de l'usuari per poder buscar els seus corresponents *holdings*.

### Requisits de Rendiment (Important):

- El sistema pot contenir desenes de milers de *Holdings*.
- Es valorarà l'eficiència algorítmica en l'eliminació de les tinences *holdings*.

**Nota:** L'ordre dels elements en `vec_holdings` **NO** és rellevant i pot alterar-se durant l'esborrat. L'ordre en `vec_users` sí que s'ha de mantenir.

### API DE CLASSES AUXILIARS

- **User**
  - `int getId()` — Retorna l'ID únic de l'usuari.
  - `string getName()` — Retorna el nom de l'usuari.
- **Asset**
  - `int getId()` — Retorna l'ID únic de l'actiu.
  - `string getAssetSymbol()` — Retorna el símbol (ex: "ETH").
- **Holding**
  - `int getUserId()` — Retorna l'ID de l'usuari propietari.
  - `int getAssetId()` — Retorna l'ID de l'actiu.

### DOCUMENTACIÓ DE REFERÈNCIA DE LA CLASSE VECTOR

Utilitzi aquesta informació per seleccionar les operacions més adequades segons els requisits d'eficiència.

- `v[i]` — Eficiència:  $O(1)$ 
  - Accedeix a l'element en la posició `i`. Permet lectura i escriptura.
  - Exemple: `int x = v[0];, v[2] = 10;`
- `v.back()` — Eficiència:  $O(1)$ 
  - Retorna una referència directa a l'últim element del vector.
  - Exemple accedir: `cout << v.back().getName();`
  - Exemple sobreescriure: `v.back() = objX;`
- `v.push_back(val)` — Eficiència:  $O(1)$ 
  - Afegeix una còpia de l'element `val` al final del vector. Augmenta la mida en 1.
  - Exemple: `v.push_back(usuari);`
- `v.pop_back()` — Eficiència:  $O(1)$ 
  - Crida al destructor de l'últim element i l'elimina del vector. Redueix la mida en 1.
  - Exemple: `v.pop_back();`
- `v.erase(it)` — Eficiència:  $O(N)$ 
  - Crida al destructor de l'element apuntat per l'iterador i l'elimina del vector. Desplaça tots els elements posteriors a l'esquerra.
  - Exemple esborrar element amb índex 2: `v.erase(v.begin() + 2);`
- `v.size()` — Eficiència:  $O(1)$ 
  - Retorna el nombre d'elements actuals.
  - Exemple: `v.size();`

## Entrada

Com a entrada hi haurà els comandaments de creació d'usuaris, actius i holdings, finalitzant amb la crida a `deleteUser`.

## Sortida

El programa principal realitzarà una comprovació automàtica "Abans i Després":

1. Imprimirà si existeix l'usuari [0,1] i quants *holdings* té [0, 1, ..., N] (ex: 1 3).
2. Executarà la funció implementada.
3. Imprimirà si l'usuari existeix [0,1] i quants *holdings* té amb aquest ID (si s'ha eliminat correctament: 0 0)

### Exemple d'entrada 1

```
crearActiu BTC Bitcoin 8
crearActiu ETH Ethereum 18
crearPreu BTC 2025-01-01 50000
crearPreu ETH 2025-01-01 3000
crearUsuari Jaume jaume@test.com 1234
afegirEuros Jaume 10000
comprar Jaume BTC 0.1
comprar Jaume ETH 1
eliminarUsuari Jaume
acabar
```

### Exemple de sortida 1

```
=====
Gestor de Carteres Cripto - CLI
=====

Seleccioneu una opció:

1. crearActiu          - Donar d'alta un nou actiu
2. crearPreu           - Afegir un preu diari a un actiu
3. mostrarActius       - Llistar tots els actius
4. mostrarPreusActiu   - Mostrar tots els preus d'un actiu

5. crearUsuari         - Donar d'alta un nou usuari
6. mostrarUsuaris      - Llistar tots els usuaris
7. crearAdreca         - Afegir una nova adreça de wallet
8. mostrarAdrecesUsuari - Mostrar totes les adreces d'un usuari

9. afegirEuros         - Afegir Euros al saldo d'un usuari
10. retirarEuros       - Retirar Euros del saldo d'un usuari

11. comprar            - Comprar un actiu per a un usuari
12. vendre             - Vendre un actiu d'un usuari
13. preuActualActiu    - Consultar l'últim preu d'un actiu
14. carteraDeUsuari    - consultar la cartera i el guany d'un usuari

15. eliminarUsuari     - Eliminar un usuari i les seves dades

0. acabar              - Sortir del programa

-----

Introduiu l'opció desitjada: Actiu BTC creat correctament
Actiu ETH creat correctament
Preu de BTC a 2025-01-01 creat correctament
Preu de ETH a 2025-01-01 creat correctament
Usuari Jaume creat correctament
10000 Euros afegits correctament a l'usuari Jaume
Compra de 0.1 unitats registrada per Jaume en BTC
Compra de 1 unitats registrada per Jaume en ETH
1 3
0 0
```

## Exemple d'entrada 2

```
crearActiu BTC Bitcoin 8
crearActiu ETH Ethereum 18
crearPreu BTC 2025-01-01 50000
crearPreu ETH 2025-01-01 3000
crearUsuari Jaume jaume@test.com 1234
crearUsuari Maria maria@test.com 5678
afegirEuros Jaume 5000
afegirEuros Maria 100000
comprar Jaume BTC 0.1
comprar Maria ETH 2
eliminarUsuari Maria
acabar
```

## Exemple de sortida 2

```
=====
Gestor de Carteres Cripto - CLI
=====
```

Seleccioneu una opció:

- |                         |   |
|-------------------------|---|
| 1. crearActiu           | - Donar d'alta un nou actiu                   |
| 2. crearPreu            | - Afegir un preu diari a un actiu             |
| 3. mostrarActius        | - Llistar tots els actius                     |
| 4. mostrarPreusActiu    | - Mostrar tots els preus d'un actiu           |
| 5. crearUsuari          | - Donar d'alta un nou usuari                  |
| 6. mostrarUsuaris       | - Llistar tots els usuaris                    |
| 7. crearAdreca          | - Afegir una nova adreça de wallet            |
| 8. mostrarAdrecesUsuari | - Mostrar totes les adreces d'un usuari       |
| 9. afegirEuros          | - Afegir Euros al saldo d'un usuari           |
| 10. retirarEuros        | - Retirar Euros del saldo d'un usuari         |
| 11. comprar             | - Comprar un actiu per a un usuari            |
| 12. vendre              | - Vendre un actiu d'un usuari                 |
| 13. preuActualActiu     | - Consultar l'últim preu d'un actiu           |
| 14. carteraDeUsuari     | - consultar la cartera i el guany d'un usuari |
| 15. eliminarUsuari      | - Eliminar un usuari i les seves dades        |
| 0. acabar               | - Sortir del programa                         |

```
-----
Introduiu l'opció desitjada: Actiu BTC creat correctament
Actiu ETH creat correctament
Preu de BTC a 2025-01-01 creat correctament
Preu de ETH a 2025-01-01 creat correctament
Usuari Jaume creat correctament
Usuari Maria creat correctament
5000 Euros afegits correctament a l'usuari Jaume
100000 Euros afegits correctament a l'usuari Maria
Compra de 0.1 unitats registrada per Jaume en BTC
Compra de 2 unitats registrada per Maria en ETH
1 2
0 0
```

## Exemple d'entrada 3

```
crearActiu BTC Bitcoin 8
crearActiu ETH Ethereum 18
crearPreu BTC 2025-01-01 50000
crearPreu ETH 2025-01-01 3000
crearUsuari Jaume jaume@test.com 1234
afegirEuros Jaume 10000
comprar Jaume BTC 0.1
comprar Jaume ETH 1
eliminarUsuari Pepe
acabar
```

## Exemple de sortida 3

```
=====
Gestor de Carteres Cripto - CLI
=====
```

Seleccioneu una opció:

- |                      |                                     |
|----------------------|-------------------------------------|
| 1. crearActiu        | - Donar d'alta un nou actiu         |
| 2. crearPreu         | - Afegir un preu diari a un actiu   |
| 3. mostrarActius     | - Llistar tots els actius           |
| 4. mostrarPreusActiu | - Mostrar tots els preus d'un actiu |
| 5. crearUsuari       | - Donar d'alta un nou usuari        |
| 6. mostrarUsuaris    | - Llistar tots els usuaris          |

7. crearAdreca	- Afegir una nova Adreca	Adreca de wallet	- Sortir del programa
8. mostrarAdrecesUsuari	- Mostrar totes les adreces d'un usuari		
-----			
9. afegirEuros	- Afegir Euros al saldo d'un usuari	Introduiu l'opció desitjada: Actiu BTC creat correctament	
10. retirarEuros	- Retirar Euros del saldo d'un usuari	Actiu BTC creat correctament	
		Preu de BTC a 2025-01-01 creat correctament	
11. comprar	- Comprar un actiu per a un usuari	Preu de BTC a 2025-01-01 creat correctament	
12. vendre	- Vendre un actiu d'un usuari	Usuari Jaume creat correctament	
13. preuActualActiu	- Consultar l'últim preu d'un actiu	Preu de BTC creat correctament	
14. carteraDeUsuari	- consultar la cartera d'un usuari	Compra de 1 unitats registrada per Jaume en BTC	
		Compra de 1 unitats registrada per Jaume en ETH	
15. eliminarUsuari	- Eliminar un usuari i les seves dades		
		0 0	

#### Exemple d'entrada 4

```

crearActiu BTC Bitcoin 8
crearUsuari Buit buit@test.com 1111
eliminarUsuari Buit
acabar

```

#### Exemple de sortida 4

```

=====
Gestor de Carteres Cripto - CLI
=====

Seleccioneu una opció:

1. crearActiu          - Donar d'alta un nou actiu
2. crearPreu           - Afegir un preu diari a un actiu
3. mostrarActius       - Llistar tots els actius
4. mostrarPreusActiu   - Mostrar tots els preus d'un actiu

5. crearUsuari         - Donar d'alta un nou usuari
6. mostrarUsuaris      - Llistar tots els usuaris
7. crearAdreca         - Afegir una nova adreça de wallet
8. mostrarAdrecesUsuari - Mostrar totes les adreces d'un usuari

9. afegirEuros         - Afegir Euros al saldo d'un usuari
10. retirarEuros        - Retirar Euros del saldo d'un usuari

11. comprar            - Comprar un actiu per a un usuari
12. vendre              - Vendre un actiu d'un usuari
13. preuActualActiu     - Consultar l'últim preu d'un actiu
14. carteraDeUsuari     - consultar la cartera i el guany d'un usuari

15. eliminarUsuari     - Eliminar un usuari i les seves dades

0. acabar              - Sortir del programa

-----
Introduiu l'opció desitjada: Actiu BTC creat correctament
Usuari Buit creat correctament
1 0
0 0

```

#### Informació del problema

Autoria : Jaume Gallego  
Generació : 2026-01-10 18:14:29

© Jutge.org, 2006–2026.  
<https://jutge.org>