

Περιγραφή κλάσεων

- Κλάση Resource:

```
public String toString(){
```

Μεθοδος που επιστρέφει την String αναπαράσταση του αντικειμένου

```
}
```

- Κλάση HumanResource:

```
public String toString(){
```

Καλεί την toString της Υπερκλάσης (Resource) και προσθέτει και επιπλέον γνωρίσματα για την συγκεκριμένη κλάση

```
}
```

- Κλάση MachineResource:

```
public String toString(){
```

Καλεί την toString της Υπερκλάσης (Resource) και προσθέτει και επιπλέον γνωρίσματα για την συγκεκριμένη κλάση

```
}
```

- Κλάση ManualMachine:

```
public String toString(){
```

Καλεί την toString της Υπερκλάσης (MachineResource) και προσθέτει και επιπλέον γνωρίσματα για την συγκεκριμένη κλάση

```
}
```

- Κλάση ProductionOrder:

```
public String toString(){
```

Επιστρεφει την string αναπαρασταση για τα αντικειμενα id και description

```
}
```

- Κλάση ConsumptionMovement:

```
public void calculateConsumptionCost(){
```

Εαν το resource ειναι HumanResource, τοτε το κοστος καταναλωσης ισουται με το κοστος καταναλωσης συν το μισθο επι το χρονο καταναλωσης δια 60.
Διαφορετικα εαν το resource ειναι AutoMachine το κοστος καταναλωσης ισουται με το κοστος καταναλωσης συν το kwh επι την σταθερα 0.089.
Διαφορετικα εαν το resource ειναι ManualMachine το κοστος καταναλωσης ισουται με το κοστος καταναλωσης συν το kwh επι την σταθερα 0.089 συν το χρονο προετοιμασιας και αποκαταστασης επι την σταθερα 0.45.

```
}
```

```
public float getConsumptionCost(){
```

Υπολογιζει το κοστος καταναλωσης καλοντας την πιο πανω μεθοδο και επιστρεφει το κοστος καταναλωσης

```
}
```

- Κλάση MainApp:

```
public void Init(){
```

Αρχικοποιει τις συλλογες resources, orders και consumptionmovements.

```
}
```

```
private void Init_Resources(){
```

Δημιουργει πορους (Ανθρωπινους και Μηχανες) και τις προσθετει στις λιστες.

```
}
```

```
private void Init_Orders(){
```

Δημιουργει εντολες και τις προσθετει στη λιστα εντολων.

```
}
```

```
private Init_ConsumptionMovements(){
```

Χρησιμοποιει μεθοδους της κλασης Random για να δημιουργισει τυχαιες κινησεις καταναλωσης.

Δημιουργει ενα τυχαιο int απο 0 μεχρι order.size() -1.

Δημιουργει ενα αλλο τυχαιο int απο 0 μεχρι 0 μεχρι resources.size()-1.

Δημιουργει εναν τυχαιο float αριθμο απο το 20 μεχρι 240.

Διαλεγει μια τυχαια εντολη παραγωγης.

Διαλεγει εναν τυχαιο πορο.

Το κανει 4 φορες για να δημιουργισει 4 τυχαιες κινησεις καταναλωσης και τα προσθετει στη λιστα κινησεων καταναλωσης.

```
}
```

```
private void showProductionOrders(){
```

Για καθε εντολη της λιστας orders τυπωνει τα μηνυματα των μεθοδων print.

```
}
```

```
private void showResources(){
```

Για καθε πορο της λιστας resources τυπωνει τα μηνυματα των μεθοδων print.

```
}
```

```
private void showConsumptionMovement(){
```

Καλει την μεθοδο showProductionOrders().

Τυπωνει μηνυμα για να διαλεξουμε εντολη

Εαν το order_index είναι μικροτερο απο 0 ή μεγαλυτερο ισο με το orders.size() τότε τυπωσε μηνυμα σφαλματος.

Για καθε κινηση της λιστας consumptionMovements, εαν οι κωδικες είναι ιδιοι τότε τυπωσε τα μηνυματα.

}

private void showMainMenu() {

Τυπωνει ολα τα μηνυματα για το κυριο μενου του προγραμματος.

}

public void run() {

Για οσο η συνθηκη while είναι true

Αρχικα καλει την μεθοδο showMainMenu.

Μετα απο την την επιλογη που θα διαλεξει ο χρηστης , για καθε περιπτωση θα καλει την αντιστοιχη μεθοδο.

Στην περιπτωση 1 την createResource

Στην περιπτωση 2 την createProductionOrder

και ουτω καθεξης για τις αλλες περιπτώσεις.

}

private void createResource(){

Τυπωνει σχετικα μηνυματα για την δημιουργια ενος πορου.

Στη πρωτη περιπτωση δημιουργει ανθρωπινο πορο.

Στη δευτερη περιπτωση δημιουργει πορο μηχανης συνεχους λειτουργιας

Στη τριτη περιπτωση δημιουργει πορο μηχανης με χρονο προετοιμασιας

}

private void createProductionOrder(){

Τυπωνει σχετικο μηνυμα

Περιμενει απο τον χρηστη να βαλει τα στοιχεια.

}

```
private void createConsumptionMovement(){
```

Καλει την μεθοδο showProductionOrders()

Περιμενει τιμη απο τον χρηστη. Εαν η τιμη ειναι μικροτερη απο 0 ή μεγαλυτερη ιση με το orders.size() τοτε θα τυπωσει μηνυμα σφαλματος.

Καλει την μεθοδο showResources()

Περιμενει τιμη απο το χρηστη. Εαν η τιμη ειναι μικροτερη απο 0 ή μεγαλυτερη ιση με το resources.size() τοτε θα τυπωσει μηνυμα σφαλματος.

Περιμενει απο τον χρηστη input για την ημερομηνια και το χρονο καταναλωσης. Δημιουργει αντικειμενο ConsumptionMovement και το προσθετει στη λιστα των κινήσεων καταναλωσης.

Τυπωνει μηνυμα για την επιτυχεις δημιουργεια του αντικειμενου.

```
}
```

```
private void calculateProductionCost(){
```

Τυπωνει σχετικο μηνυμα.

Καλει την μεθοδο showProductionOrder()

Περιμενει τιμη απο τον χρηστη. Εαν η τιμη ειναι μικροτερη απο 0 ή μεγαλυτερη ιση με το orders.size() τοτε θα τυπωσει μηνυμα σφαλματος.

Για καθε κινήση στη λιστα consumptionMovements εαν τα id ειναι ιδια τοτε το κοστος παραγωγης θα ισουται με το κοστος παραγωγης συν το κοστος καταναλωσης.

Τυπωνει το αποτελεσμα.

```
}
```

```
private void calculateResourceConsumptionCost(){
```

Τυπωνει σχετικο μηνυμα

Καλει την μεθοδο showResources()

Περιμενει τιμη απο τον χρηστη. Εαν η τιμη ειναι μικροτερη απο 0 ή μεγαλυτερη ιση με το resources.size() τοτε θα τυπωσει μηνυμα σφαλματος. Για καθε κινήση στη λιστα consumptionMovements εαν τα id ειναι ιδια τοτε το κοστος καταναλωσης του πορου θα ισουται με το κοστος καταναλωσης του πορου συν το κοστος καταναλωσης.

Τυπωνει το αποτελεσμα.

```
}
```