Загальна постановка :  
 У відповідному фрагменті коду необхідно на мисти коментаря дописати фрагмент коду теким чином, щоб виконати завдання, записане у коментарі.

1. **static def** newElemList(n, g) {  
    n%4 + g(n+1)  
    }  
   **def** ls=[]  
   *//створіть 10 елементів списка за допомогою newElemList*

*10.times { number -> ls.add(newElemList(10, { it + number })) }*println ls

1. **def** list = [45, 12, -9, 8, 14, -89, 4, -23, 56, 458, 1, -12]  
   **def** newList=list.drop(1)  
   *//необхідно додати до списку newList перший, останній та //середній елементи зі списку list*

*def newList = list.drop(1).dropRight(1)*

*newList.removeAt((newList.size() / 2) as int)*println newList

1. **def** list = [45, 12, -9, 8, 14, -89, 4, -23, 56, 458, 1, 123, -23, 12, 6]  
    **def newList = list**

**.collate(3)**

**.collect { it.min() as Integer }***//необхідно змінити правило формування елементів списку   
//так, щоб елемент нового списку був*

*//мінімальним елементом серед кажних трьох  
//що слідують послідовно один за одним у списку list*println newList

1. **def** ls = [1, -2, 8]  
   **def** Size = 5  
     
   Size.downto(1)**{** *//перетворіть код додавання нових елементів у список так,  
   //щоб кажен другий доданий елемент у список був числом 55* ls.add(ls.sum()/(it%2?:10) **as** Integer)

ls.add(55)  
**}**println ls

1. **def** list = [45, 12, -9, 8, 1, -89, 4, -23, 56, 458, 1, 123, -23, 12, 6]  
     
   *//змініть код створення нового списку так, щоб кожні п“ять  
   // елементів списку були видсортовані у спадному порядку,  
   // але являлися елементами усього списку, а не створених //підсписків***def** ls = list.collect **{** it **}  
    .collate(5)**

**.collect { it.sort {-it} }**println list  
println ls

## **class** Human { String toString() { **"**$**{**name**} was born** $**{**age**} years ago"** } **def name** = **"NoName" def age** = 0

## void setAge(age) {

## if (this.age < age && 0 < age )

## this.age = age

## } } **def** chel = **new** Human(**name**: **"Vasya Pupkin"**, **age**: 21) println chel *//перетворіьб код так, щоб змінити значення віку //об“єкта типу Human на відємне число або значення //менше ніж те, що містить властивість age, було неможливо* chel.setAge(-15) println chel

1. **class** Human {  
    String toString() {  
    **"**$**{**name**} was born** $**{**age**} years ago"** }  
     
    **def name** = **"Bot"  
    def age** = new Date()  
   }  
     
   **def** chel = **new** Human(**name**: **"Vasya Pupkin"**, **age**: 21)  
   println chel  
   **def** incognito = **new** Human()

println incognito  
*//перетворіть код так, щоб замість імені NoName*

*// об'єкта типу Human, створеного по дефолту,*

*// на екран виводилося ім'я Bot з віком, що дорівнює*

*// значенням поточного дня*

1. **class** Human {  
    String toString() {  
    **"**$**{**name**} was born** $**{**age**} years ago"** }  
     
    **def name** = **"NoName"  
    def age** = 0  
   }

**def** people = [**new** Human(**name**: **"Vasya Pupkin"**, **age**: 21),

**new** Human(),

**new** Human(**age**: 40),

**new** Human(**name**: **"Uncle Goga"**),

**new** Human(**name**: **"Aunt Sonya"**, **age**: 60),

**new** Human(**name**: **"It's me"**)]

**def** notBot= people.findAll { it.name != "NoName" && it.age != 0}

*// тут повинен бути код, який дозволить створити*

*// список тих Human, які повністю не визначені, тобто*

*// якісь з властивостей містять дефолтні значення*

println **"Все, что создано"**

**for**(el **in** people){

println el

}

println **"Реально люди:"**

**for**(el **in** realPaople ){

println el

}

1. **class** Human {  
    String toString() {  
    **"**$**{**name**} was born** $**{**age**} years ago"** }  
     
    **def name** = **"NoName"  
    def age** = 0  
   }  
     
   **def** people = [**new** Human(**name**: **"Vasya Pupkin"**, **age**: 21),  
    **new** Human(),

**new** Human(**age**: 40),

**new** Human(**name**: **"Uncle Goga"**),

**new** Human(**name**: **"Aunt Sonya"**, **age**: 60),

**new** Human(**name**: **"It's me"**)]

**def notBot= people.findAll {it.age > 30}**

*// тут повинен бути код, який дозволить створити*

*// список тих Human, яким вже більше 30 років*

println **"Все, что создано"**

**for**(el **in** people){

println el

}

println **"Те, кому за 30:"**

**for**(el **in** notBot){

println el

}

1. **class** Human {  
    String toString() {  
    **"**$**{**name**} was born** $**{**age**} years ago"** }  
     
    **def name** = **"NoName"  
    def age** = 0  
   }

static def CreateNewList(){

def answer = ''

def list = []

while (answer != 'no'){

println "Enter name"

def name = System.in.newReader().readLine()

println "Enter age"

def age = System.in.newReader().readLine() as int

list.add(new Human(name: name, age: age))

println "Create new human?"

answer = System.in.newReader().readLine()

}

return list

}

**def** people = [**new** Human(**name**: **"Vasya Pupkin"**, **age**: 21),  
 **new** Human(),

**new** Human(**age**: 40),

**new** Human(**name**: **"Uncle Goga"**),

**new** Human(**name**: **"Aunt Sonya"**, **age**: 60),

**new** Human(**name**: **"It's me"**)]

**def notBot= CreateNewList()**

*// тут повинен бути код, який дозволить створити*

*// новий список об'єктів Human, в якому запропоновано*

*// користувачам ввести нові значення для дефолтних властивостей*

println **"Все, что создано"**

**for**(el **in** people){

println el

}

println **"Создано и дополнено:"**

**for**(el **in** notBot){

println el

}