Міністерство освіти і науки України

Національний Технічний Університет України

«Київський Політехнічний Інститут»

Навчально-науковий комплекс

«Інститут прикладного системного аналізу»

Кафедра системного проектування

Лабораторна робота №2

з дисципліни «Теорія прийняття рішень»

на тему:« Прийняття рішень в умовах ризику»

Варіант №14

Виконав: студент 3 курсу

групи ДА-52

Плотніков С.О.

Перевірив: Безносик О.Ю.

Київ 2018

**Мета роботи:** ознайомитись з методами прийняття рішень в умовах, коли вибір деякої стратегії пов’язаний з певним набором станів середовища з визначеною ймовірністю.

**Завдання для варіанту №14**

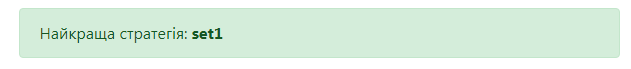
Таблиця 1

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| № | Місто | І | ІІ | ІІІ | ІV | V | VI | VII | VIII | IX | X | XI | XII |
| 14 | Урумчі (Китай) | -13 | -11 | -1 | +11 | +18 | +23 | +25 | +24 | +18 | +9 | -2 | -10 |

**Виконання роботи**

1. Визначити найкращу стратегію при поверненні протягом одного з 12-ти місяців за умови, що ймовірність повернення в кожен з місяців однакова.





1. Визначити найкращу стратегію за при поверненні протягом одного сезону за наданих наборів ймовірностей повернення у кожен з місяців.

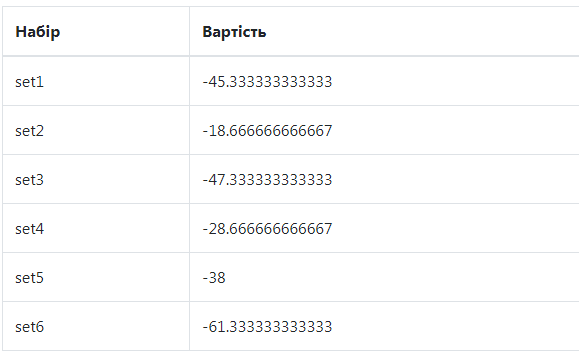
ЗИМА:

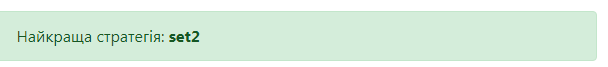
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| №№ | Назва набору | Місяць, Pi | | | | | | | | | | | |
| 01 | 02 | 03 | 04 | 05 | 06 | 07 | 08 | 09 | 10 | 11 | 12 |
| 1 | Зима | 1/3 | 1/3 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1/3 |
| 2 | Весна | 0 | 0 | 1/3 | 1/3 | 1/3 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 3 | Літо | 0 | 0 | 0 | 0 | 0 | 1/3 | 1/3 | 1/3 | 0 | 0 | 0 | 0 |
| 4 | Осінь | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1/3 | 1/3 | 1/3 | 0 |





ВЕСНА:



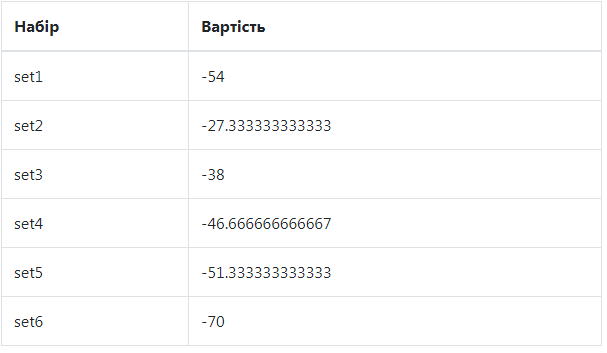


ЛІТО:





ОСІНЬ:





1. Визначити найкращу стратегію при поверненні протягом одного з 12-ти місяців за умови, що ймовірність повернення взимку втричі більша за інші місяці.



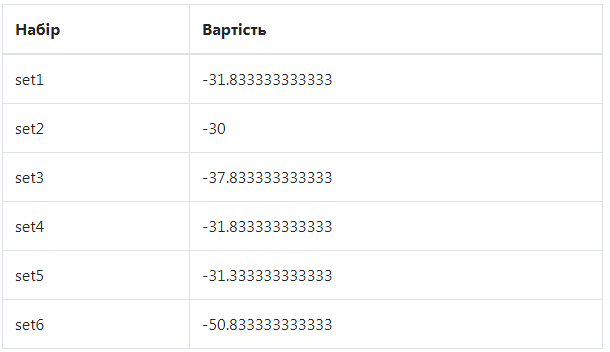


1. Визначити найкращу стратегію при поверненні протягом одного з 12-ти місяців за умови, що ймовірність повернення залежить від кількості днів у місяці (рік вважати не високосним).





1. Визначити найкращу стратегію при поверненні протягом одного з 12-ти місяців за умови, що ймовірність повернення в кожен з місяців однакова, а початкова вартість речей з номерами № 2, 4, 9, 15, 16 зменшилася втричі.





Таблиця 2. Результати визначення найкращих стратегій

|  |  |  |  |
| --- | --- | --- | --- |
| № | Назва завдання | Найкраща стратегія(-ї), *xi* | Значення |
| 1 | За 12-ть місяців | **Set1** | -35.83333 |
| 2 | «Зима» | **Set1** | 0 |
| 3 | «Весна» | **Set2** | -18.666666667 |
| 4 | «Літо» | **Set5** | 0 |
| 5 | «Осінь» | **Set2** | -27.3333333 |
| 6 | «Зима» х3 | **Set1** | -36.01 |
| 7 | За 12-ть місяців (за кількістю днів) | **Set1** | -36.01 |
| 8 | За 12-ть місяців (зменш. вартість) | **Set2** | -30 |

**Висновок:** найкраща стратегія(найкращий комплект речей) є перший набір(set1), але при зменшенні вартості ціни в 3 рази на деякі речі кращою стратегією стає набір №2(set2)

**Лістинг програми:**

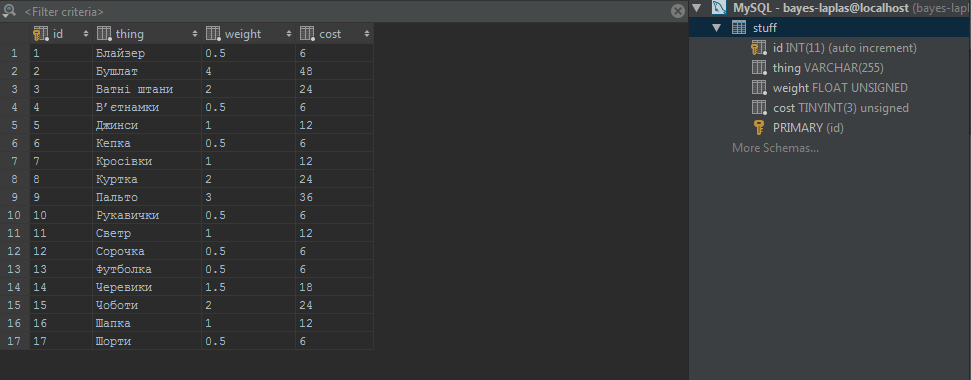
**loadSetOfStuff.php**

Містить масив наборів речей

**<?php***/\*\*  
 \* Created by PhpStorm.  
 \* User: Serhii  
 \* Date: 28.03.2018  
 \* Time: 23:05  
 \*/***return** [  
 ['temperature' => ['min' => *PHP\_INT\_MIN*, 'max' => -10],  
 'stuff' => ['Шапка', 'Бушлат', 'Рукавички', 'Ватні штани', 'Чоботи'], 'weight' => 9.5],  
 ['temperature' => ['min' => -9, 'max' => 0],  
 'stuff' => ['Шапка', 'Пальто', 'Рукавички', 'Джинси', 'Чоботи'], 'weight' => 7.5],  
 ['temperature' => ['min' => 1, 'max' => 10],  
 'stuff' => ['Кепка', 'Куртка', 'Джинси', 'Черевики'], 'weight' => 5],  
 ['temperature' => ['min' => 11, 'max' => 20],  
 'stuff' => ['Светр', 'Джинси', 'Кросівки'], 'weight' => 3],  
 ['temperature' => ['min' => 21, 'max' => 30],  
 'stuff' => ['Блайзер', 'Сорочка', 'Джинси', 'Кросівки'], 'weight' => 3],  
 ['temperature' => ['min' => 31, 'max' => *PHP\_INT\_MAX*],  
 'stuff' => ['Блайзер', 'Футболка', 'Шорти', 'В’єтнамки'], 'weight' => 2]  
];

**Табл. Stuff**

Містить речі, які можуть входити в набори



**index.php**

**<?php***/\*\*  
 \* Created by PhpStorm.  
 \* User: Serhii  
 \* Date: 28.03.2018  
 \* Time: 21:41  
 \*/***declare**(strict\_types = 1);  
**include\_once** 'vendor/autoload.php';  
  
$decision = **new** MakingDecisions();  
  
//загружаем наборы одежды в виде массива  
$sets = **include**('loadSetOfStuff.php');  
  
//кол-во наборов одежды  
$countOfSets = count($sets);  
//Получаем массив наборов в виде объектов  
$setsOfStuff = [];  
$index = 1;  
**foreach** ($sets **as** $set) {  
 $stuffObjectsArray = [];  
 **for** ($i = 0; $i < count($set['stuff']); $i++) {  
 **for** ($j = 0; $j < count($decision->getStuff()); $j++) {  
 **if** ($set['stuff'][$i] == $decision->getStuff()[$j]->getThing()) {  
 $stuffObjectsArray[$set['stuff'][$i]] = $decision->getStuff()[$j];  
 }  
 }  
 }  
 $setsOfStuff["set{$index}"] = **new** SetOfStuff($set['temperature'], $stuffObjectsArray, $set['weight']);  
 $index++;  
};  
  
$decision->setSetsOfStuff($setsOfStuff);  
$decision->fillGoodsTable($setsOfStuff);  
  
//Построим таблицу необходимых товаров по каждому месяцу для каждого товара  
$tableOfGoods = $decision->getGoodsTable();  
**?>** <!doctype html>  
 <html lang="en">  
 <head>  
 <meta charset="UTF-8">  
 <meta name="viewport"  
 content="width=device-width, user-scalable=no, initial-scale=1.0, maximum-scale=1.0, minimum-scale=1.0">  
 <meta http-equiv="X-UA-Compatible" content="ie=edge">  
 <title>Document</title>  
 <link rel="stylesheet" href="vendor/twbs/bootstrap/dist/css/bootstrap.min.css">  
 <script src="vendor/twbs/bootstrap/dist/js/bootstrap.min.js"></script>  
 </head>  
 <body>  
 <div class="container-fluid">  
 <div class="row">  
 <div class="col-sm-12">  
 <table class="table table-bordered">  
 <thead>  
 <tr>  
 <th scope="col">Набір</th>  
 <th scope="col">Вартість</th>  
 **<?php  
 foreach** ($decision->getMonths() **as** $month => $temperature):**?>** <th scope="col">**<?=** $month **?>**</th>  
 **<?php endforeach**; **?>** </tr>  
 </thead>  
 <tbody>  
 **<?php  
 foreach** ($tableOfGoods **as** $nameOfSet => $set): **?>** <tr>  
 <td>**<?=** $nameOfSet; **?>**</td>  
 <td>**<?=** $set['freightCost'] **?>**</td>  
 **<?php foreach** ($set['months'] **as** $n => $month): **?>** <td>  
 **<?php  
 if** (is\_null($month)) :  
 **echo** '---';  
 **else** :  
 **?>** <table>  
 **<?php foreach** ($month['stuff'] **as** $name => $item): **?>** <tr>  
 <td>**<?=** $name **?>**</td>  
 </tr>  
 **<?php endforeach**; **?>** <tr>  
 <td><i>Вартість</i>: <b>**<?=** $month['cost'] **?>**</b></td>  
 </tr>  
 </table>  
 **<?php endif**; **?>** </td>  
 **<?php endforeach**; **?>** </tr>  
 **<?php endforeach**; **?>** </tbody>  
 </table>  
 </div>  
 </div>  
 <div class="row">  
 <div class="col-sm-12">  
 <h5>Найкраща стратегія при поверненні протягом одного з 12-ти місяців за умови, що ймовірність  
 повернення  
 в кожен з місяців однакова</h5>  
 </div>  
 </div>  
 <div class="row">  
 <div class="col-sm-6">  
 **<?php** /\*  
 \* Визначити найкращу стратегію при поверненні протягом  
 \* одного з 12-ти місяців за умови, що ймовірність повернення в кожен з місяців однакова  
 \*/  
 **list**($setOfStrategies, $bestStrategy) = $decision->getStrategyWithEqualProbability();  
 **?>** <table class="table table-bordered">  
 <thead>  
 <tr>  
 <th>Набір</th>  
 <th>Вартість</th>  
 </tr>  
 </thead>  
 <tbody>  
 **<?php  
 foreach** ($setOfStrategies **as** $name => $cost): **?>** <tr>  
 <td>**<?=** $name **?>**</td>  
 <td>**<?=** $cost **?>**</td>  
 </tr>  
 **<?php endforeach**; **?>** </tbody>  
 </table>  
 </div>  
 <div class="col-sm-6">  
 <div class="alert alert-success">Найкраща стратегія: <b>**<?=** $bestStrategy['strategyName'] **?>**</b></div>  
 </div>  
 </div>  
 <div class="row">  
 <div class="col-sm-12">  
 <h5>Найкраща стратегія при поверненні протягом одного сезону за наданих наборів ймовірностей  
 повернення у кожен з місяців. ЗИМА</h5>  
 </div>  
 </div>  
 <div class="row">  
  
 <div class="col-sm-6">  
 **<?php** /\*  
 \* Визначити найкращу стратегію при поверненні протягом одного сезону за наданих наборів ймовірностей  
 \* повернення у кожен з місяців. ЗИМА  
 \*/  
 $availableMonths = ['Січень', 'Лютий', 'Грудень'];  
 **list**($setOfStrategies, $bestStrategy) = $decision->getStrategyInSeason($availableMonths);  
 **?>** <table class="table table-bordered">  
 <thead>  
 <tr>  
 <th>Набір</th>  
 <th>Вартість</th>  
 </tr>  
 </thead>  
 <tbody>  
 **<?php  
 foreach** ($setOfStrategies **as** $name => $cost): **?>** <tr>  
 <td>**<?=** $name **?>**</td>  
 <td>**<?=** $cost **?>**</td>  
 </tr>  
 **<?php endforeach**; **?>** </tbody>  
 </table>  
 </div>  
 <div class="col-sm-6">  
 <div class="alert alert-success">Найкраща стратегія: <b>**<?=** $bestStrategy['strategyName'] **?>**</b></div>  
 </div>  
 </div>  
 <div class="row">  
 <div class="col-sm-12">  
 <h5>Найкраща стратегія при поверненні протягом одного сезону за наданих наборів ймовірностей  
 повернення у кожен з місяців. ВЕСНА</h5>  
 </div>  
 </div>  
 <div class="row">  
 <div class="col-sm-6">  
 **<?php** /\*  
 \* Визначити найкращу стратегію при поверненні протягом одного сезону за наданих наборів ймовірностей  
 \* повернення у кожен з місяців. Весна  
 \*/  
 $availableMonths = ['Березень', 'Квітень', 'Травень'];  
 **list**($setOfStrategies, $bestStrategy) = $decision->getStrategyInSeason($availableMonths);  
 **?>** <table class="table table-bordered">  
 <thead>  
 <tr>  
 <th>Набір</th>  
 <th>Вартість</th>  
 </tr>  
 </thead>  
 <tbody>  
 **<?php  
 foreach** ($setOfStrategies **as** $name => $cost): **?>** <tr>  
 <td>**<?=** $name **?>**</td>  
 <td>**<?=** $cost **?>**</td>  
 </tr>  
 **<?php endforeach**; **?>** </tbody>  
 </table>  
 </div>  
 <div class="col-sm-6">  
 <div class="alert alert-success">Найкраща стратегія: <b>**<?=** $bestStrategy['strategyName'] **?>**</b></div>  
 </div>  
 </div>  
  
 <div class="row">  
 <div class="col-sm-12">  
 <h5>Найкраща стратегія при поверненні протягом одного сезону за наданих наборів ймовірностей  
 повернення у кожен з місяців. ЛІТО</h5>  
 </div>  
 </div>  
 <div class="row">  
 <div class="col-sm-6">  
 **<?php** /\*  
 \* Визначити найкращу стратегію при поверненні протягом одного сезону за наданих наборів ймовірностей  
 \* повернення у кожен з місяців. Літо  
 \*/  
 $availableMonths = ['Червень', 'Липень', 'Серпень'];  
 **list**($setOfStrategies, $bestStrategy) = $decision->getStrategyInSeason($availableMonths);  
 **?>** <table class="table table-bordered">  
 <thead>  
 <tr>  
 <th>Набір</th>  
 <th>Вартість</th>  
 </tr>  
 </thead>  
 <tbody>  
 **<?php  
 foreach** ($setOfStrategies **as** $name => $cost): **?>** <tr>  
 <td>**<?=** $name **?>**</td>  
 <td>**<?=** $cost **?>**</td>  
 </tr>  
 **<?php endforeach**; **?>** </tbody>  
 </table>  
 </div>  
 <div class="col-sm-6">  
 <div class="alert alert-success">Найкраща стратегія: <b>**<?=** $bestStrategy['strategyName'] **?>**</b></div>  
 </div>  
 </div>  
 <div class="row">  
 <div class="col-sm-12">  
 <h5>Найкраща стратегія при поверненні протягом одного сезону за наданих наборів ймовірностей  
 повернення у кожен з місяців. ОСІНЬ</h5>  
 </div>  
 </div>  
 <div class="row">  
 <div class="col-sm-6">  
 **<?php** /\*  
 \* Визначити найкращу стратегію при поверненні протягом одного сезону за наданих наборів ймовірностей  
 \* повернення у кожен з місяців. Осінь  
 \*/  
 $availableMonths = ['Вересень', 'Жовтень', 'Листопад'];  
 **list**($setOfStrategies, $bestStrategy) = $decision->getStrategyInSeason($availableMonths);  
 **?>** <table class="table table-bordered">  
 <thead>  
 <tr>  
 <th>Набір</th>  
 <th>Вартість</th>  
 </tr>  
 </thead>  
 <tbody>  
 **<?php  
 foreach** ($setOfStrategies **as** $name => $cost): **?>** <tr>  
 <td>**<?=** $name **?>**</td>  
 <td>**<?=** $cost **?>**</td>  
 </tr>  
 **<?php endforeach**; **?>** </tbody>  
 </table>  
 </div>  
 <div class="col-sm-6">  
 <div class="alert alert-success">Найкраща стратегія: <b>**<?=** $bestStrategy['strategyName'] **?>**</b></div>  
 </div>  
 </div>  
  
 <div class="row">  
 <div class="col-sm-12">  
 <h5>Найкраща стратегія при поверненні протягом одного з 12-ти місяців за умови, що ймовірність  
 повернення взимку втричі більша за інші місяці.</h5>  
 </div>  
 </div>  
 <div class="row">  
 <div class="col-sm-6">  
 **<?php** /\*  
 \* Визначити найкращу стратегію при поверненні протягом одного з 12-ти місяців за умови, що ймовірність  
 \* повернення взимку втричі більша за інші місяці.  
 \*/  
 **list**($setOfStrategies, $bestStrategy) = $decision->getStrategyWithGreaterWinterProbability();  
 **?>** <table class="table table-bordered">  
 <thead>  
 <tr>  
 <th>Набір</th>  
 <th>Вартість</th>  
 </tr>  
 </thead>  
 <tbody>  
 **<?php  
 foreach** ($setOfStrategies **as** $name => $cost): **?>** <tr>  
 <td>**<?=** $name **?>**</td>  
 <td>**<?=** $cost **?>**</td>  
 </tr>  
 **<?php endforeach**; **?>** </tbody>  
 </table>  
 </div>  
 <div class="col-sm-6">  
 <div class="alert alert-success">Найкраща стратегія: <b>**<?=** $bestStrategy['strategyName'] **?>**</b></div>  
 </div>  
 </div>  
  
 <div class="row">  
 <div class="col-sm-12">  
 <h5>Найкраща стратегія при поверненні протягом одного з 12-ти місяців за умови, що ймовірність  
 повернення залежить від кількості днів у місяці .</h5>  
 </div>  
 </div>  
 <div class="row">  
 <div class="col-sm-6">  
 **<?php** /\*  
 \* Визначити найкращу стратегію при поверненні протягом одного з 12-ти місяців за умови, що ймовірність  
 \* повернення залежить від кількості днів у місяці (рік вважати не високосним).  
 \*/  
 **list**($setOfStrategies, $bestStrategy) = $decision->getStrategyWithMonthlyProbability();  
 **?>** <table class="table table-bordered">  
 <thead>  
 <tr>  
 <th>Набір</th>  
 <th>Вартість</th>  
 </tr>  
 </thead>  
 <tbody>  
 **<?php  
 foreach** ($setOfStrategies **as** $name => $cost): **?>** <tr>  
 <td>**<?=** $name **?>**</td>  
 <td>**<?=** $cost **?>**</td>  
 </tr>  
 **<?php endforeach**; **?>** </tbody>  
 </table>  
 </div>  
 <div class="col-sm-6">  
 <div class="alert alert-success">Найкраща стратегія: <b>**<?=** $bestStrategy['strategyName'] **?>**</b></div>  
 </div>  
 </div>  
 <div class="row">  
 <div class="col-sm-12">  
 <h5>Найкраща стратегія при поверненні протягом одного з 12-ти місяців за умови, що ймовірність  
 повернення в кожен з місяців однакова, а початкова вартість речей з номерами  
 № 2, 4, 9, 15, 16 зменшилася втричі.</h5>  
 </div>  
 </div>  
 <div class="row">  
 <div class="col-sm-6">  
 **<?php** /\*  
 \* Визначити найкращу стратегію при поверненні протягом одного з 12-ти місяців за умови, що ймовірність повернення  
 \* в кожен з місяців однакова, а початкова вартість речей з номерами № 2, 4, 9, 15, 16 зменшилася втричі.  
 \*/  
 **list**($setOfStrategies, $bestStrategy) = $decision->getStrategyWithReducedCost();  
 **?>** <table class="table table-bordered">  
 <thead>  
 <tr>  
 <th>Набір</th>  
 <th>Вартість</th>  
 </tr>  
 </thead>  
 <tbody>  
 **<?php  
 foreach** ($setOfStrategies **as** $name => $cost): **?>** <tr>  
 <td>**<?=** $name **?>**</td>  
 <td>**<?=** $cost **?>**</td>  
 </tr>  
 **<?php endforeach**; **?>** </tbody>  
 </table>  
 </div>  
 <div class="col-sm-6">  
 <div class="alert alert-success">Найкраща стратегія: <b>**<?=** $bestStrategy['strategyName'] **?>**</b></div>  
 </div>  
 </div>  
 </div>  
 </body>  
 </html>

**MakingDecisions.php**

Містить методи вибору найкращого набору

**<?php***/\*\*  
 \* Created by PhpStorm.  
 \* User: Serhii  
 \* Date: 29.03.2018  
 \* Time: 16:59  
 \*/*//Класс принятия решений  
**class** MakingDecisions  
{  
 //соединение с бд  
 **private** $pdo;  
 //набор доступных вещей  
 **private** $stuff;  
 //комплекты вещей  
 **private** $setsOfStuff = [];  
 //температура в Урумчи  
 **private** $months = ['Січень' => -13, 'Лютий' => -11, 'Березень' => -1, 'Квітень' => 11, 'Травень' => 18,  
 'Червень' => 23, 'Липень' => 25, 'Серпень' => 24, 'Вересень' => 18, 'Жовтень' => 9, 'Листопад' => -2,  
 'Грудень' => -10];  
 //кол-во месяцов  
 **private** $countOfMonths;  
 //стоимость перевозки 1 вещи  
 **private** $freightCostPerOneThing = 10;  
 //доп. стоимость покупки одной дополнительной вещи  
 **private** $costOfOneAdditionalThing = 2;  
  
 //таблица, отображающая, что необходимо докупить для каждого набора по каждому месяцу  
 **private** $goodsTable;  
  
 **public function** \_\_construct()  
 {  
 $this->pdo = (**new** DB())->getPdo();  
 //получаем набор вещей  
 $this->getStuffFromDb();  
 //кол-во месяцов  
 $this->countOfMonths = count($this->months);  
 }  
  
 */\*\*  
 \* Установка комплектов вещей  
 \** ***@param*** *array $setsOfStuff  
 \*/* **public function** setSetsOfStuff(**array** $setsOfStuff)  
 {  
 $this->setsOfStuff = $setsOfStuff;  
 }  
  
 //получение набора вещей с бд  
 **private function** getStuffFromDb():void  
 {  
 $sql = "SELECT \* FROM stuff";  
 $stmt = $this->pdo->query($sql);  
 $this->stuff = $stmt->fetchAll(PDO::*FETCH\_CLASS*, 'Thing');  
 **return**;  
 }  
  
 //Заполняем таблицу вещей, которые необходимо докупить  
 **public function** fillGoodsTable(**array** $setsOfStuff)  
 {  
 */\*\*  
 \** ***@param*** *SetOfStuff $set  
 \*/* **foreach** ($setsOfStuff **as** $nameOfSet => $set) {  
 $this->goodsTable[$nameOfSet] = [];  
 //стоимость перевозки  
 $freightCost = $this->freightCostPerOneThing \* $set->getWeight();  
 $this->goodsTable[$nameOfSet]['freightCost'] = $freightCost;  
 //min and max температура ношения даного набора вещей  
 $minTemperature = $set->getTemperature()['min'];  
 $maxTemperature = $set->getTemperature()['max'];  
 //Рассчитаем по каждому месяцу  
 **foreach** ($this->months **as** $month => $temperature) {  
 /\*Если температура ношения нашего набора входит в промежуток температуры данного месяца,  
 то значит ничего нам докупать не надо\*/  
 **if** (($minTemperature <= $temperature) && ($temperature <= $maxTemperature)) {  
 $this->goodsTable[$nameOfSet]['months'][$month] = **null**;  
 } **else** {  
 //Находим набор вещей, которые соответствуют температуре текущего месяца  
 $staffForCurrentTemperature = **null**;  
 **foreach** ($setsOfStuff **as** $setOfStuff) {  
 **if** (($setOfStuff->getTemperature()['min'] <= $temperature) && ($temperature <= $setOfStuff->getTemperature()['max'])) {  
 $staffForCurrentTemperature = $setOfStuff->getStuff();  
 **break**;  
 }  
 }  
 //Находим недостающие вещи для данной температуры данного месяца  
 $additionalStuff = array\_diff\_key($staffForCurrentTemperature, $set->getStuff());  
  
 //Рассчитаем стоимость недостающих элементов  
 $countOfAdditionalStuff = count($additionalStuff);  
 $sum = $countOfAdditionalStuff \* $this->costOfOneAdditionalThing;  
 **foreach** ($additionalStuff **as** $item) {  
 $sum += $item->getCost();  
 }  
 $this->goodsTable[$nameOfSet]['months'][$month]['stuff'] = $additionalStuff;  
 $this->goodsTable[$nameOfSet]['months'][$month]['cost'] = $sum;  
 }  
 }  
 }  
 }  
  
 **public function** getStuff()  
 {  
 **return** $this->stuff;  
 }  
  
 */\*\*  
 \** ***@return*** *array  
 \*/* **public function** getGoodsTable():**array** {  
 **return** $this->goodsTable;  
 }  
  
 */\*\*  
 \** ***@return*** *array  
 \*/* **public function** getMonths(): **array** {  
 **return** $this->months;  
 }  
  
 //Вычисляет наилучшую стратегию со всех возможных(лучший набор вещей)  
 **private function** getBestStrategy(**array** $valuesOfStrategies):**array** {  
 $bestStrategyName = '';  
 $bestStrategyValue = *PHP\_INT\_MIN*;  
 **foreach** ($valuesOfStrategies **as** $nameOfStrategy => $valueOfStrategy) {  
 **if** ($bestStrategyValue < $valueOfStrategy) {  
 $bestStrategyName = $nameOfStrategy;  
 $bestStrategyValue = $valueOfStrategy;  
 }  
 }  
 **return** ['strategyName' => $bestStrategyName, 'strategyValue' => $bestStrategyValue];  
 }  
  
 /\*  
 \* Возвращает наилучшую стратегию при возвращении на протяжении 1 с 12-и месяцев, за условием, что вероятность  
 \* возвращения в каждом месяце одинакова  
 \*/  
 **public function** getStrategyWithEqualProbability():**array** {  
 $probability = 1 / $this->countOfMonths;  
 //Значения для каждой стратегии  
 $valuesOfStrategies = [];  
 **foreach** ($this->goodsTable **as** $numOfSet => $set) {  
 $sum = 0;  
 **foreach** ($set['months'] **as** $month) {  
 **if** (!is\_null($month)) {  
 $sum -= ($month['cost'] \* $probability);  
 }  
 }  
 $valuesOfStrategies[$numOfSet] = $sum;  
 }  
 //найдем лучшую стратегию(лучший набор вещей)  
 $bestStrategy = $this->getBestStrategy($valuesOfStrategies);  
 **return** [$valuesOfStrategies, $bestStrategy];  
 }  
  
  
 /\*  
 \* Возвращает наилучшую стратегию при возвращении напротяжении одного сезона(зима-весна-лето-осень)  
 \*/  
 **public function** getStrategyInSeason(**array** $months):**array** {  
 $probability = 1 / 3;  
 $valuesOfStrategies = [];  
 $availableMonths = $months;  
 **foreach** ($this->goodsTable **as** $numOfSet => $set) {  
 $sum = 0;  
 **foreach** ($set['months'] **as** $monthName => $month) {  
 **if** (!is\_null($month) && in\_array($monthName, $availableMonths)) {  
 $sum -= ($month['cost'] \* $probability);  
 }  
 }  
 $valuesOfStrategies[$numOfSet] = $sum;  
 }  
 //найдем лучшую стратегию(лучший набор вещей)  
 $bestStrategy = $this->getBestStrategy($valuesOfStrategies);  
 **return** [$valuesOfStrategies, $bestStrategy];  
 }  
  
 /\*  
 \* Возвращает наилучшую стратегию при возвращении на протяжении 1 с 12-и месяцев, за условием, что вероятность  
 \* возвращения зимой в 3 раза выше чем в другие месяца  
 \*/  
 **public function** getStrategyWithGreaterWinterProbability():**array** {  
 $winterProbability = 0.166;  
 $probability = 0.055;  
 $valuesOfStrategies = [];  
 $winterMonths = ['Січень', 'Лютий', 'Грудень'];  
 **foreach** ($this->goodsTable **as** $numOfSet => $set) {  
 $sum = 0;  
 **foreach** ($set['months'] **as** $monthName => $month) {  
 **if** (!is\_null($month) && in\_array($monthName, $winterMonths)) {  
 $sum -= ($month['cost'] \* $winterProbability);  
 } **elseif** (!is\_null($month)) {  
 $sum -= ($month['cost'] \* $probability);  
 }  
 }  
 $valuesOfStrategies[$numOfSet] = $sum;  
 }  
 //найдем лучшую стратегию(лучший набор вещей)  
 $bestStrategy = $this->getBestStrategy($valuesOfStrategies);  
 **return** [$valuesOfStrategies, $bestStrategy];  
 }  
  
 /\*  
 \* Возвращает наилучшую стратегию при возвращении на протяжении одного з 12-ти месяцев при уcловии, что вероятность  
 \* возвращения зависит от количества дней в месяце(год считаем не высокосным)  
 \*/  
 **public function** getStrategyWithMonthlyProbability():**array** {  
 $probabilityFor28Days = 0.077;  
 $probabilityFor30Days = 0.082;  
 $probabilityFor31Days = 0.085;  
  
 $valuesOfStrategies = [];  
 $monthsWith28Days = ['Лютий'];  
 $monthsWith30Days = ['Квітень', 'Червень', 'Вересень', 'Листопад'];  
 $monthsWith31Days = ['Січень', 'Березень', 'Травень', 'Липень', 'Серпень', 'Жовтень', 'Грудень'];  
 **foreach** ($this->goodsTable **as** $numOfSet => $set) {  
 $sum = 0;  
 **foreach** ($set['months'] **as** $monthName => $month) {  
 **if** (!is\_null($month) && in\_array($monthName, $monthsWith28Days)) {  
 $sum -= ($month['cost'] \* $probabilityFor28Days);  
 } **elseif** (!is\_null($month) && in\_array($monthName, $monthsWith30Days)) {  
 $sum -= ($month['cost'] \* $probabilityFor30Days);  
 } **elseif** (!is\_null($month) && in\_array($monthName, $monthsWith31Days)) {  
 $sum -= ($month['cost'] \* $probabilityFor31Days);  
 }  
 }  
 $valuesOfStrategies[$numOfSet] = $sum;  
 }  
 //найдем лучшую стратегию(лучший набор вещей)  
 $bestStrategy = $this->getBestStrategy($valuesOfStrategies);  
 **return** [$valuesOfStrategies, $bestStrategy];  
 }  
  
 /\*  
 \* Возвращает наилучшую стратегию при возвращении на протяжении одного з 12-ти месяцев при уcловии, что вероятность  
 \* возвращения в каждом месяце одинакова, а начальная стоимость вещей с номерами № 2, 4, 9, 15, 16 уменьшится втрое  
 \*/  
 **public function** getStrategyWithReducedCost():**array** {  
  
 $itemIds = [2, 4, 9, 15, 16];  
 //Уменьшаем стоимость вещей с указаным id  
 **foreach** ($this->stuff **as** &$item) {  
 **if** (in\_array($item->getId(), $itemIds)) {  
 $item->setCost($item->getCost() / 3);  
 }  
 }  
 //Заполним таблицу, в которой будет отображено, что необходимо докупить для каждого набора по каждому месяцу  
 $this->fillGoodsTable($this->setsOfStuff);  
  
 **return** $this->getStrategyWithEqualProbability();  
  
 }  
}

**SetOfStuff.php**

Класс описания набора вещей

**<?php***/\*\*  
 \* Created by PhpStorm.  
 \* User: Serhii  
 \* Date: 28.03.2018  
 \* Time: 22:32  
 \*/***class** SetOfStuff  
{  
 **private** $weight;  
 **private** $stuff = [];  
 **private** $temperature = [];  
  
 **public function** \_\_construct(**array** $temperature, **array** $stuff, float $weight)  
 {  
 $this->temperature = $temperature;  
 $this->stuff = $stuff;  
 $this->weight = $weight;  
 }  
  
 */\*\*  
 \** ***@return*** *array  
 \*/* **public function** getStuff(): **array** {  
 **return** $this->stuff;  
 }  
  
 */\*\*  
 \** ***@return*** *float  
 \*/* **public function** getWeight(): float  
 {  
 **return** $this->weight;  
 }  
  
 */\*\*  
 \** ***@return*** *array  
 \*/* **public function** getTemperature(): **array** {  
 **return** $this->temperature;  
 }  
}

**Thing.php**

Класс описания конкретной вещи

**<?php***/\*\*  
 \* Created by PhpStorm.  
 \* User: Serhii  
 \* Date: 28.03.2018  
 \* Time: 22:13  
 \*/*//Класс, отвечающий за вещь с набора  
**class** Thing  
{  
 **private** $id;  
 **private** $thing;  
 **private** $weight;  
 **private** $cost;  
  
 */\*\*  
 \** ***@return*** *mixed  
 \*/* **public function** getId()  
 {  
 **return** $this->id;  
 }  
  
 */\*\*  
 \** ***@return*** *mixed  
 \*/* **public function** getCost()  
 {  
 **return** $this->cost;  
 }  
  
 */\*\*  
 \** ***@param*** *float $cost  
 \*/* **public function** setCost(float $cost)  
 {  
 $this->cost = $cost;  
 }  
  
 */\*\*  
 \** ***@return*** *mixed  
 \*/* **public function** getWeight()  
 {  
 **return** $this->weight;  
 }  
  
 */\*\*  
 \** ***@return*** *mixed  
 \*/* **public function** getThing()  
 {  
 **return** $this->thing;  
 }  
}

**DB.php**

Класс подключения к базе данных

**<?php***/\*\*  
 \* Created by PhpStorm.  
 \* User: Serhii  
 \* Date: 29.03.2018  
 \* Time: 17:04  
 \*/***class** DB  
{  
 **private** $pdo;  
 **private const** *HOST* = 'localhost';  
 **private const** *DBNAME* = 'bayes-laplas';  
 **private const** *USER* = 'root';  
 **private const** *PASSSWORD* = '21091992';  
  
 **function** \_\_construct()  
 {  
 $this->pdo = **new** PDO('mysql:host=' . **self**::*HOST* . ';dbname=' . **self**::*DBNAME* . '', **self**::*USER*, **self**::*PASSSWORD*);  
 }  
  
 */\*\*  
 \** ***@return*** *PDO  
 \*/* **public function** getPdo(): PDO  
 {  
 **return** $this->pdo;  
 }  
}