

2.3	Данные класса	16
2.3.1	id	16
2.3.2	name	16
3	класс Base, delete method	16
3.1	метод (self)	16
3.1.1	delete method()	16
3.1.2	delete method()	16
		16
		16
		16
		16

	.11.3.2 name	3
.12	ла B sC ne. Idel.Panamu n u	3
.12.1	н т кт (n)	36
	.12.1.1 Panamu n u () [1/k]	36
	.12.1.2 Panamu n u () [/k]	36
	.12.1.3 Panamu n u () [k/k]	36
.12.2	ет <u>н</u>	36
	.12.2.1 Exlsts()	36
	.12.2.2 enenateld()	37
	.12.2.3 a e ata()	37
.12.3	Данные класса	37
	.12.3.1 a tual alue	37
	.12.3.2 ntl alEmu n	37
	.12.3.3 ld	37
	.12.3.4 n mlnal	37
	.12.3. tuNum	37
	.12.3.6 unlt	38
.13	ла B sC ne. Idel.P st	38
.13.1	н т кт (n)	38
	.13.1.1 P st() [1/k]	38
	.13.1.2 P st() [/k]	38
	.13.1.3 P st() [k/k] .12.2 . ет <u>н</u>	39

.1	ла	BaseC ne. del.Requirements	42
.1.1	н т кт (su)		42
.1.1.1		Requirements() [1/k]	42
.1.1.2		Requirements() [/k]	42
.1.1.3		Requirements() [k/k]	43
.1.2	ет us		43
.1.2.1		Exists()	43
.1.2.2		enateld()	43
.1.2.3		ate ata()	43

.18.1.3 ta ()

Элементы не ечи лени

Idle	
------	--

umf В sC ue. tatusC dle [stn n e

Элементы не ечи лени

allde

|

ЛАВА

ЛАВЫ

5.1 ЛСС SysC.e.M.eI.Challenge

Открытые члены

- Challenge()
- Challenge(Pdu t n du t C ntm llObje t ntm llObje t ChallengeT e hallen eT e ta sta ll atI n l atI n ateTime be InTime ateTime endTime stIn des nI tI n=null)
- Challenge(Int Id)
- Int enenateId()
- Id a e ata (QllteC nne tI n nne tI n)

Открытые тр уты

- Int Id
- Pdu t n du t
- C ntm llObje t ntm llObje t
- ChallengeT e hallen eT e
- ta sta
- ll atI n l atI n
- ateTime be InTime
- ateTime endTime
- stIn des nI tI n

ll Кон труктор ы)

1.1.1.2 Challenge() [k]

```

1 by CowMod I.Cltting.Cltting (
2     Produ produ ,
3     ComsolObj comsolObj ,
4     Cltting typ Cltting typ ,
5     ff ff,
6     fo fo fo fo,
7     return b gntion ,
8     return ndtion ,
9     string d wtp fo = null ) [false]

```

1.1.1.3 Challenge() [k/k]

```

1 by CowMod I.Cltting.Cltting (
2     in id ) [false]

```

1.2 Методы

1.2.1 generateId()

```

1 by CowMod I.Cltting . new Id () [false]

```

1.2.2 generate()

```

void 1 by CowMod I.Cltting . v 1 (

```

3.2 challenge

`ChallengeType` — это `enum` `ChallengeType` из `enum` `ChallengeType`

3.3 contentType

`ContentType` — это `enum` `ContentType` из `enum` `ContentType`

3.4 description

`description` — это `string` из `string` `description`

3.5 endTime

`endTime` — это `int` из `int` `endTime`

3.6 id

`id` — это `int` из `int` `id`

3.7 location

`location` — это `string` из `string` `location`

3.8 product

`product` — это `string` из `string` `product`

3.9 status

`status` — это `int` из `int` `status`

Описание и определение членов в классе на языке Python

- `del Challenge.s`

3.2.3 а е ата()

```
void D y Cou .Mod l.Comm founIn af . v D (
    Qll Comm foun onn foun ) [fllln ]
```

3.3 Д нные кл

3.3.1 Id

```
ln D y Cou .Mod l.Comm founIn af .Id
```

3.3.2 name

```
ling D y Cou .Mod l.Comm founIn af .ln nn
```

О явления и пи ания член в кла а на ят я в фа ле

- del C nne tl nInterfa e. s

5.4 л сс SysC e.M el.C nt ll ject

Открытые члены

- C ntn llObje t ()
- C ntn llObje t (stln name stln serialNumber stln ensl n stln aent)
- C ntn llObje t (ln Id)
- ln enerateId ()
- Id а е ата (QllteC nne tl n nne tl n)

Открытые т т че к е члены

- statl b l Exlsts (stln serialNumber)

Открытые тр уты

- ln Id
- stln name
- stln serialNumber
- stln ensl n
- stln aent

Ст т ч е к е открытые д нные

- statl QiteC nne tl n usersC nne tl n
- statl QiteC nne tl n dum C nne tl n

крытые т т ч е к е члены

- statl

2.3 Be InChallen e()

```

    if (u.Cod == y.Cod.Cod) {
        glnCln.Hng (
            wng.podu == ,
            wng.o == ,
            wng.o.at.I.unmb == ,
            wng.oV.u.fom,
            wng.oP.u.m,
            wng.H.Hng.Hyp == ,
            wng.lo.fom == ,
            wng.d.atp.fom == "" ) [talln ], [ t ]
    }
}

```

Инициали и ет н в е и пштаніе.

2.4 Cl seC nne t! ns()

```
void D_y_Conn_Clo_Conn_tom ( ) [init], [f ], [priv ]
```

Зак ~~н~~ивает в е ~~т~~к ~~н~~т~~н~~е ~~е~~щ~~н~~ения.

2. CmdPn ess()

```

1  void D y Com .Com .ComPro (
2      string pp,
3      string ug ,
4      bool logCom ol = f l ,
5      bool logLogg u = f l ) [ l i n e ] , [ c o l ]

```

ет для выполнения требований в перечисленных агентах.

2.6 um en e()

```

i u Cod D y Coi .Coi .DumpDug (
    uing d ,
    uing[ i u ) [initia] , [ i ]

```

. 2.9 Exe Q₁()

! u Cod

2.1.1. `enableAll()`

```
void CCom::EnableAll(
    QCom::Communication,
    QString) {
    // ...
}
```

метод `enableAll()` для включения всех устройств и файла на входе с помощью параметров.

2.1.6. `RetrieveAllUsers()`

```
void CCom::RetrieveAllUsers(
    QString) {
    // ...
}
```

метод `RetrieveAllUsers()` для получения всех пользователей и файла в папке `Program`.

2. Конструкторы

7.2.1.1. `Emil::Emil()` [1/3]

```
Emil::Emil() {
```

7.2.1.2. `Emil::Emil()` [2/3]

```
Emil::Emil(const std::string& login,
            const std::string& password,
            const std::string& email) {
```

7.2.1.3. `Emil::Emil()` [3/3]

```
Emil::Emil(const std::string& login) {
```

3. Методы

7.2.1.1. `Exists()`

```
bool Emil::Exists(const std::string& login) const {
```

7.2.1.2. `ExtraData()`

```
std::vector<std::string> Emil::ExtraData(const std::string& login) const {
```

4. Данные кл

.8. ат n() [k]

5.10 л сс SysC e.M el.Meth l 5M

Открытые члены
etho ()TJ 0 g 0 0 F48 8.9663 Tf 1- 0 20-12.833Td [(•)]TJ 0 g 0 G 0 0 1 rg 0 0 1 RG48 8.9663 Tf 16.1 0 20 Td [(etho)-278dology

•

5.11 Л сс SysC e.M el.M e

Открытые члены

- `de()`
- `de(stln name)`
- `de(int id)`
- `int enewateld()`
- `id a e ata (QuteC nne tl n nne tl n)`

Открытые т т ч е к е члены

- `statl b l Exlsts (stln name)`

Открытые тр уты

- `int id`
- `stln name`

Кон труктор ы)

.11.1.1 `de()` [1/к]

by Coe .Mod l.Mod .Mod () [11111]

.11.1.2 `de()` [/к]

by Coe .Mod l.Mod .Mod (
 sing on n) [11111]

.11.1.3 `de()` [к/к]

by Coe .Mod l.Mod .Mod (
 in id) [11111]

2 Методы

Открытые т т ч е к е члены

- stat**l** b **l** **Exists** (stat**ln** **tuNum**)

Открытые тр уты

- **Int** **Id**
- de **lnal** **n** **mdnal**
- de **lnal** **nttl** **alEm** **n**
- de **lnal** **a** **tual** **alue**
- stat**ln** **unlt**
- stat**ln** **tuNum**

¶2 ¶ Кон труктор ы)

.¶2.¶.¶ Раван **n** u () [¶/¶]

¶ y Cou .¶¶od **l.P** **n** **m** **ro¶¶/P** **n** **m** **roup** () [¶¶¶¶]
[¶/¶]

u u

4.1.2. Р. ду т([/x]

4.3.2 name

```
using DependencyComposer.Mod.IProductItem
```

Объявления и объявления член в классе на **заканчиваются** в файле

- `del Products`

5.15 класс SysC e.M el.Requirements

Открытые члены

- `Requirements()`
- `Requirements(string name, string id, Number)`
- `Requirements(int id)`
- `int GenerateId()`
- `id Generate(Queue<int> n, int n)`

Открытые статические члены

- `static bool Exists(string name)`

Открытые свойства

- `int id`
- `string name`
- `string id, Number`

Конструкторы

```
1 Requirements() [1/5]
```

```
2 using DependencyComposer.Mod.IProductItem, DependencyComposer.Mod.IProductItem [1/5]
```

```
3 Requirements() [1/5]
```

```
4 using DependencyComposer.Mod.IProductItem, DependencyComposer.Mod.IProductItem (
    string name,
    string id, Number) [1/5]
```


3.3 name

using D y Coo .Mod L. ufa m m .m m

О явления и пи ания член в кла а на ят я в фа ле

- del Requirements. s

5.15 л сс SysC e.Sessl n

Открытые т т че к е члены

- statl Id O en ()
- statl statl n etUserName ()
- statl sessl n ata enerate efault sessl n ata ()
- statl b lls. edln ()
- statl User nants et nants ()
- statl statl n nants tnl n (User nants nants)
- statl b l Reque ne nants

6.1.1 Auth()

```

1  u Code 0 y Conn. form. u (
2      log login,
3      log password) [login], [ ]

```

Авт и и ет п ль вателя п пе еанном л гин и па лю. В в ащает 0 е ли авт и ация п -

6.6 etUserName()

```
if (String.IsNullOrEmpty(userName)) [return], [return]
```

Возвращает фамилию пользователя или "unknown" если пользователь не авторизован.

Формат вывода: Фамилия И.

6.7 wants to join()

```
if (String.IsNullOrEmpty(userName)) [return], [return]
    if (String.IsNullOrEmpty(userName)) [return], [return]
```

Возвращает текстовое представление пользователя.

```
"domain" | "operation" | "user" | "username"
```

6.8 isLoggedIn()

```
if (bool.IsNullOrEmpty(userName)) [return], [return]
```

Возвращает true если пользователь не авторизован.

6.9 Logout()

16.1.1.1 Requirements()

```
bool D::Requirements::operator==(const Requirements& r) const {
    return (*this == r);
}
```

Возвращает true, если два текста являются перестановками.

16.1.1.2 IsCommentChallenge()

```
bool D::IsCommentChallenge(const string& s) const {
    return s.find('#') != string::npos;
}
```

Сверяет текст с шаблоном. Если текст совпадает с шаблоном, то возвращает true, иначе false.

Если текст является комментарием, то возвращает true, иначе false.

Возвращает false, если API не поддерживается.

5.17 Структур SysC e.Session t

Открытые тр уты

- Em l en em l en
- Pl nam tate n nam tate
- stIn filename
- Challenge a tleChallenge
- Ist < Test naml > a tleTests

СтруктурПоддонное такуей е

екле 210ДенноВ келно т те тов

ОПолное прее веднны те тов

екле 123000Дтеое

n 11D2 de eond ene

20.3.1. `__init__`

`__init__` — метод класса, который вызывается при создании объекта.

20.3.2. `__del__`

`__del__` — метод класса, который вызывается при уничтожении объекта.

20.3.3. `__str__`

`__str__` — метод класса, который возвращает строковое представление объекта.

20.3.4. `__repr__`

`__repr__` — метод класса, который возвращает строковое представление объекта, которое можно использовать для его повторного создания.

20.3.5. `__len__`

`__len__` — метод класса, который возвращает длину объекта.

20.3.6. `__getitem__`

`__getitem__` — метод класса, который возвращает элемент объекта по индексу.

20.3.7. `__setitem__`

`__setitem__` — метод класса, который устанавливает элемент объекта по индексу.

20.3.8. `__iter__`

`__iter__` — метод класса, который возвращает итератор объекта.

Описание и использование членов класса можно найти в файле

- `Test_tutorial.py`

.21.2.3 et ash()

и ing Ю y Cou . ll . ll ll (

Глава 6

Файлы

3.1 Файл C e.cs

Классы

- `lass B sC ne.C ne`

Протиптичен

- `names а е B sC ne`

3.2 Файл Challenge.cs

Классы

- `lass B sC ne. en`

Протиптичен

- `names а е B sC ne`

3.3 Файл M el/Ch llenge.cs

Классы

- `lass B sC ne. del.Challen e`

Протиптичен

- `names а е B sC ne`
- `names а е B sC ne. del`

§.12 Файл M el/P m up.cs

Класс

Иерархия

- enum B { sC en2mm name tate { B { sC en2mm name tate.Idle = 0 B { sC en2mm name tate. nkin
= 1 B { sC en2Pn name tate.TestIn = 2 } }
- enum B { sC en2User nants { { te.Tesene
teO = enat ne B 9, C en2User nant.An

Exists

B sC ne. del.Challen eT e 14
 B sC ne. del.C nne tl nIntenfa e 16
 B sC ne. del.C ntn HObje t 18
 B sC ne. del. e antment 2
 B sC ne. del.Em l en 27
 B sC ne. del. atl n 29
 B sC ne. del. eth d l 32
 B sC ne. del. de 34
 B sC ne. del.Panam n u 36
 B sC ne. del.P st 39
 B sC ne. del.Pn du t 41
 B sC ne. del.Requhements 43
 B sC ne. del. ta 0

Extra tAll

B sC ne. del. e antment 2
 B sC ne. del.Em l en 27
 B sC ne. del.P st 39
 B sC ne. del. ta 0

file

B sC ne. en 31

filename

B sC ne. essi n 47
 B sC ne. essi n ata 48

firstName

B sC ne. del. ta 1

enerate efault essi n ata

B sC ne. essi n 4

enerateFile Name

B sC ne. en 31

enerateId

B sC ne. del.Challen e 12
 B sC ne. del.Challen eT e 1
 B sC ne. del.C nne tl nIntenfa e 16
 B sC ne. del.C ntn HObje t 18
 B sC ne. del. e antment 2
 B sC ne. del. atl n 29
 B sC ne. del. eth d l 33
 B sC ne. del. de 3
 B sC ne. del.Panam n u 36
 B sC ne. del.P st 39
 B sC ne. del.Pn du t 41
 B sC ne. del.Requhements 43
 B sC ne. del. ta 0
 B sC ne. del.Test naml 3
 B sC ne. del.Test tatl

et nants

B sC ne. essi n 4

et ash

B sC ne. Utils 7

etTests

B sC ne. del.Test naml 3

etUserName

B sC ne. essi n 4

nantsIn n en

B sC ne 8

nants tath

B sC ne. essi n 46
 nantsUnath nl ed
 B sC ne 8

ash

B sC ne. Utils 8

Id

B sC ne. del.Challen e 13
 B sC ne. del.Challen eT e 1
 B sC ne. del.C nne tl nIntenfa e 17
 B sC ne. del.C ntn HObje t 19
 B sC ne. del. e antment 26
 B sC ne. del. atl n 29
 B sC ne. del. eth d l 33
 B sC ne. del. de 3
 B sC ne. del.Panam n u 37
 B sC ne. del.P st 39
 B sC ne. del.Pn du t 41
 B sC ne. del.Requhements 43
 B sC ne. del. ta 1
 B sC ne. del.Test naml 4
 B sC ne. del.Test tatl 6

Idle

B sC ne 8

Initial eC nne tl n

B sC ne. C ne 22

Initial eUser ataC nne tl n

B sC ne. C ne 22

Is edIn

B sC ne. essi n 46

ad tatl Tests

B sC ne. C ne 22

atl n

B sC ne. del. atl n 28 29

l atl n

B sC ne. del.Challen e 13

ad

B sC ne. en 31

en. s 9

In

B sC ne. C ne 22

l In

B sC ne. del.Em l en 27

In Ahead Auth nl ed

B sC ne 8

In alld In

B sC ne 8

In alldPass

B sC ne 8

In N Re lstenedUsers

B sC ne 8

ut

B sC ne. C ne 22

B sC ne. essi n 46

en e tiles

B sC ne. C ne 22

```

    eth d l
    B sC ne. del. eth d l 32
meth d l
    B sC ne. del.Test tatl

```

B sC ne.C ne 23
 tatus
 B sC ne.C ne 23
 status
 B sC ne. del.Test naml 4
 tatusC de
 B sC ne 8
 tatusC de. s 64
 surname
 B sC ne. del. ta 1

 Test
 B sC ne.C ne 23
 Test naml
 B sC ne. del.Test naml 2
 Testen
 B sC ne 8
 TestIn
 B sC ne 8
 Test tatl
 B sC ne. del.Test tatl
 test tatl
 B sC ne. del.Test naml 4
 tsIndex
 B sC ne. del.Test tatl 6
 tuNum
 B sC ne. del.Panam n u 37

 unt
 B sC ne. del.Panam n u 37
 Useu nants
 B sC ne 8
 usensC nne tl n
 B sC ne.C ne 24
 usensfilename
 B sC ne.C ne 24
 Utlls. s 64

 enlf ash
 B sC ne.Utlls 8
 ensl n
 B sC ne. del.C ntn llObje t 19

 nkln
 B sC ne 8