

BPO

TD 3 - Octet, Adresse et Message

reseau.adresses

Class Octet

java.lang.Object
reseau.adresses.Octet

```
public class Octet  
extends java.lang.Object
```

Field Summary

Fields

Modifier and Type	Field and Description
protected boolean[]	bits

Constructor Summary

Constructors

Constructor and Description
Octet() Tous les bits sont à 0
Octet(int entier) Constructeur à partir d'un entier
Octet(Octet o) Constructeur de copie (copie profonde)

Method Summary

All Methods	Static Methods	Instance Methods	Concrete Methods
Modifier and Type	Method and Description		
void	ajouter(int i) Ajouter i		
boolean	estUneLettre() Retourne vrai si la valeur de l'octet est le code d'une lettre		
boolean	estUnPoint() Retourne vrai si la valeur de l'octet est le code du caractère point		
int	getValue() Retourne la valeur décimale de l'octet		
void	inverser() Complément à 1		
static void	main(java.lang.String[] args)		
void	masquer(Octet masque) Application d'un masque		
boolean	plusPetitQue(int i) Retourne vrai si l'octet est inférieur à i		
void	setUn() Fixer la valeur de l'octet avec tous les bits à 1		
void	setUn(int nbUn) Fixer la valeur de l'octet : nombre de bits à 1 (en commençant par le bit de poids le plus fort)		
java.lang.String	toString() Retourne une chaîne avec la valeur entière de l'octet		

```
public class Adresse
extends java.lang.Object
```

Constructor Summary

Constructors
Constructor and Description
Adresse (Adresse a)
Constructeur de copie (copie profonde)
Adresse (int nbBits)
Adresse avec tous les bits à 0
Adresse (int nbBits, int nbBitsUn)
Adresse masque composée de nbBitsUn bits à 1 suivis de 0 pour compléter.
Adresse (Octet... al)
Constructeur à partir du tableau d'octets
Adresse (java.lang.String s)
Constructeur à partir d'une adresse écrite sous forme de notation décimale pointée (par ex : 245.156.0.0)

Method Summary

All Methods	Instance Methods	Concrete Methods
Modifier and Type	Method and Description	
int	getNbreOctets ()	
	Retourne le nombre d'octets	
Octet	getOctet (int k)	
	Consulter un des octets de l'adresse, k est le rang de l'octet, k ≥ 0	
void	inverser ()	
	Inverser les octets : complément à 1 de chaque octet (0 → 1, 1 → 0)	
void	masquer (Adresse masque)	
	Appliquer un masque	
void	setOctet (Octet o, int k)	
	Fixer un des octets de l'adresse, k est le rang de l'octet, k ≥ 0	
void	setOctets (Octet... alo)	
	Fixer les octets	
int	size ()	
	Retourne le nombre de bits	
java.lang.String	toString ()	
	Construit la chaîne de caractères où la valeur de chaque octet est séparé du suivant par un point.	

Class Message

java.lang.Object
reseau.Message

```
public class Message
extends java.lang.Object
```

Constructor Summary

Constructors	
Constructor and Description	
<code>Message()</code>	Constructeur d'un message vide
<code>Message(Adresse adr)</code>	Constructeur d'un message à partir de l'adresse
<code>Message(int... v)</code>	Constructeur d'un message à partir des entiers (1 entier est codé sur 2 octets)
<code>Message(Message mess)</code>	Constructeur de copie (copie profonde)
<code>Message(short... v)</code>	Constructeur d'un message à partir des petits entiers (1 petit entier est codé sur un seul octet)
<code>Message(java.lang.String mot)</code>	Constructeur d'un message à partir de la chaîne de caractères

Method Summary

All Methods		Instance Methods	Concrete Methods
Modifier and Type	Method and Description		
void	<code>ajouter(Adresse adr)</code>	Ajouter les octets d'une adresse à la fin	
void	<code>ajouter(int x)</code>	Ajouter un entier à la fin, entier ≥ 0	
void	<code>ajouter(Message mess)</code>	Concaténer un autre message	
void	<code>ajouter(Octet o)</code>	Ajouter un octet à la fin	
void	<code>ajouter(short x)</code>	Ajouter un petit entier à la fin, entier ≥ 0	
void	<code>augmenter(int i, int bi, int bs)</code>	Augmenter de i chaque octet dont la valeur est comprise entre les valeurs bi et bs	
Adresse	<code>extraireAdresse(int nbOctets)</code>	Extraire les nbOctets premiers octets pour en faire une adresse	
java.lang.String	<code>extraireChaine()</code>	Transformer le message en une suite de lettres, si possible	
int	<code>extraireEntier(int index)</code>	Extraire les 2 octets situés en index et index+1 pour en faire un entier octets forts puis faibles (big endian)	
Octet	<code>getOctet(int index)</code>	Retourne l'octet d'indice i dans le Message	
int	<code>size()</code>	Retourne le nombre d'octets	
void	<code>supprimer(int i)</code>	On enlève les i premiers éléments	
void	<code>supprimer(int debut, int fin)</code>	On enlève les éléments entre debut et fin inclus	
java.lang.String	<code>toString()</code>		

Écrire une séquence d'instructions (extraites d'un main) qui :

- Crée l'octet représentant l'entier 68, et vérifie s'il s'agit ou non du code d'une lettre.
- Crée l'adresse de 32 bits commençant par 8 bits à 1 et les autres à 0, puis l'inverse avant de l'afficher.
- Crée l'adresse 245.156.0.0 et vérifie que le second octet est inférieur à 200.
- Crée le message avec les entiers 101 et 110 puis lui ajoute à la fin l'entier 19 avant de l'afficher.
- Extrait du message précédent le dernier entier.
- Crée le message à partir de l'adresse 245.156.0.0 créée précédemment, puis ajoute le message "Une adresse" ; puis extrait l'adresse du message ainsi constitué.