



Examen Java de base 20/12/2014 Variante A

NOM:
PRENOM:
GROUPE : ABDTW G
Durée <u>2H</u> Documents <u>non autorisés</u>

Questions de cours 7P					
1. Dor	ner le rôle de mots réservés suivants				
this()					
gunor					
super.					
static					
abstract					
extends					
<u>en cenab</u>					
2. Que	elles sont les règles à respecter lors de la redéfinition d'une méthode				
3. Cite	er deux rôles de l'utilisation des interfaces en java ?				
	,				
<i>4</i> . Déc	rire en quelques lignes la gestion des exceptions en java ?				
5 0	and la mot régaryé « abetract » est utilisé en Java ?				
J. Qua	ınd le mot réservé « abstract » est utilisé en Java ?				



CODE 3



Examen Java de base 20/12/2014 Variante A

NOM :	
PRENOM:	
GROUPE : ABDTW G	
Durée 2H Documents non autorisés	

Exercice 1 : Donner l'affichage issu de l'exécution des codes java (compilables) 8 P CODE 1 public class Document { Affichage CODE1: public static void m() { System.out.println("Document"); public static void main(String[] args) { Document d1=new Livre(); Livre d2=new Livre(); d1.m();...... d2.m();...... class Livre extends Document(..... public static void m() { System.out.println("Livre"); CODE 2 public class Sport { **Affichage CODE2:** public Sport() { System.out.println("Sport"); public static void main(String[] args) { FootBall f = new FootBall(90); class SportFoot extends Sport { private int duree; public SportFoot(int duree) { this.duree = duree; System.out.println(this.duree++); class FootBall extends SportFoot { public FootBall(int duree) { super (duree); System.out.println(""+duree+duree);





Examen Java de base 20/12/2014 Variante A

NOM :
PRENOM:
GROUPE : ABDTW G
Durée 2H Documents non autorisés

public class Livre (Affichage CODE3:
<pre>private int nbPages;</pre>	Militage Cobes.
<pre>private String titre;</pre>	
<pre>public Livre(int nbPages) {</pre>	
<pre>this("" + nbPages);</pre>	
<pre>this.nbPages = nbPages; }</pre>	
<pre>public Livre(String titre) {</pre>	
this.titre += titre;	
}	
<pre>public Livre(int nbPages, String titre) {</pre>	
<pre>this(++nbPages);</pre>	
<pre>this.nbPages += nbPages;</pre>	
<pre>this.titre += titre;</pre>	
}	
<pre>public static void main(String[] args) {</pre>	
Livre 1 = new Livre(100, "Java");	
System.out.println(1.nbPages); System.out.println(1.titre);	
)	
}	
CODE 4	
public class Voiture (Affichage CODE4 :
<pre>private static int nbVoitures; private static String categorie;</pre>	
parado de la companya	
<pre>public Voiture(int nb) {</pre>	
this("" + nb);	
nbVoitures += nb;	
}	
<pre>public Voiture(String cat) { categorie += cat;</pre>	
)	
•	
<pre>public static void main(String[] args) {</pre>	
Voiture 11 = new Voiture (50);	
Voiture 12 = new Voiture(50); System.out.println(11.categorie+11.nbVoitures);	
System.out.println(12.categorie+12.nbVoitures);	
}	
}	
CODE 5	
<pre>public static void main(String[] args) {</pre>	Affichage CODE5:
if (true) (
for (; false true;) (
System.out.println("true");	
}	
}	
	1





Examen Java de base 20/12/2014 Variante A

NOM:
PRENOM:
GROUPE : ABDTW G
Durée <u>2H</u> Documents <u>non autorisés</u>

Exercice 2: Y-a-t-il des erreurs de compilation? Expliquer? (5 P)					
(les erreurs de compilation suivantes : la casse, omission de « ; », omission des « {}»ou «() » n'existent					
pas dans cet exercice)					
CODE 1					
<pre>public class Test {</pre>	□ compile sans erreurs				
<pre>public static void method(){</pre>	1				
private int valeur=20;					
<pre>System.out.println(valeur); }</pre>	☐ Ne compile pas				
}					
<u> </u>					
CODE 2					
public class Test {	□ compile sans erreurs				
<pre>final int valeur; {</pre>					
valeur=10;					
}	☐ Ne compile pas				
}					
CODE 3					
public abstract class Test {	□ compile sans erreurs				
final abstract int method();	La compne sans erreurs				
}					
	☐ Ne compile pas				
CODE 4					
public class Test {	□ compile sans erreurs				
<pre>public void method(){</pre>					
ì					
}	☐ Ne compile pas				
class Test1 extends Test{					
<pre>protected void method(){</pre>					
}					
CODE C					
<pre>CODE 5 public class Test {</pre>	acmaile cane amoune				
<pre>public static void main(String[] args) {</pre>	□ compile sans erreurs				
try {					
int d = 20 / 0;	☐ Ne compile pas				
$\}$ catch (Exception e) $\{$	— 110 compile pas				
<pre>} catch (ArithmeticException e) {</pre>					
, catch (militalmetronacepoion c) (
}					
}					
}					