|  |  |  |
| --- | --- | --- |
| java2\_week12\_기본API | 학번 : 20237107 | 이름 : 하태영 |

* **개념 확인**

1. **[생성형 AI 활용]** Object 클래스의 멤버 메소드를 사용하여 동일한 객체를 판단하는 방법을 제시하세요

|  |  |
| --- | --- |
| 생성형 AI | perplexity |
| 질문 | Object 클래스의 멤버 함수를 사용하여 객체 내용을 비교할 수 있는 방법을 제시하세요 |
| 답변 | Java에서 Object 클래스의 멤버 메소드를 활용해 객체 동일성을 판단하는 방법은 다음과 같습니다. 기본적으로 equals() 메소드는 객체의 \*\*참조값(주소)\*\*을 비교하지만, 필요에 따라 **내용 기반 비교**로 재정의할 수 있습니다. 1. **기본** equals() **메소드의 동작**  * Object 클래스의 equals()는 == 연산자와 동일하게 **참조값 비교**를 수행합니다[1](https://kephilab.tistory.com/92)[3](https://docs.oracle.com/javase/tutorial/java/IandI/objectclass.html)[5](https://developer-yeony.tistory.com/107).   java  Object obj1 = **new** Object();  Object obj2 = **new** Object();  **boolean** result = obj1.equals(obj2); *// false (서로 다른 객체)*   * 두 객체가 **물리적으로 동일한 메모리 주소**를 가리킬 때만 true를 반환합니다.  2. **내용 기반 비교를 위한** equals() **오버라이드** 객체의 **논리적 동등성**을 판단하려면 equals()를 재정의해야 합니다. 예를 들어 Member 클래스의 id 필드가 같으면 동일한 객체로 취급하려면 다음과 같이 구현합니다[1](https://kephilab.tistory.com/92)[4](https://www.baeldung.com/java-comparing-objects):  java  @Override  **public** **boolean** equals(Object obj) {  **if** (**this** == obj) **return** true; *// 동일 참조면 true*  **if** (obj == **null** || getClass() != obj.getClass()) **return** false; *// 타입 검사*  Member member = (Member) obj; *// 타입 변환*  **return** id.equals(member.id); *// id 필드 비교*  } 주요 구현 단계:  1. **동일 참조 확인**: this == obj로 빠른 반환[4](https://www.baeldung.com/java-comparing-objects). 2. **null 및 타입 검사**: instanceof 또는 getClass() 사용[1](https://kephilab.tistory.com/92)[4](https://www.baeldung.com/java-comparing-objects). 3. **필드 비교**: 핵심 필드의 값을 비교합니다.  3. hashCode() **메소드 오버라이드**  * equals()를 재정의할 때는 반드시 hashCode()도 함께 재정의해야 합니다[2](https://www.tutorialspoint.com/java/lang/object_equals.htm)[4](https://www.baeldung.com/java-comparing-objects). (동일한 객체는 동일한 해시코드를 가져야 하기 때문입니다.)   java  @Override  **public** **int** hashCode() {  **return** Objects.hash(id); *// id 필드를 기반으로 해시코드 생성*  } 4. Objects.equals()**를 이용한 null-safe 비교** Java 7부터 도입된 java.util.Objects 클래스의 equals()는 **null을 안전하게 처리**합니다[4](https://www.baeldung.com/java-comparing-objects):  java  PersonWithEquals joe = **new** PersonWithEquals("Joe", "Portman");  PersonWithEquals joeAgain = **new** PersonWithEquals("Joe", "Portman");  **boolean** result = Objects.equals(joe, joeAgain); *// true (내용 비교)*   * 두 객체가 모두 null이면 true, 하나만 null이면 false를 반환합니다. * 내부에서 equals()를 호출하므로 재정의된 로직을 사용합니다.  비교 방법 정리  | **방법** | **비교 기준** | **사용 사례** | | --- | --- | --- | | == 연산자 | 참조값 | 물리적 동일성 확인 (기본 타입 변수 비교) | | equals() (재정의 전) | 참조값 | 기본 객체 비교 (의미 없는 경우多) | | equals() (재정의 후) | 필드 값 | 사용자 정의 논리적 동등성 판단 (예: id 비교) | | Objects.equals() | null-safe 비교 | null 가능성이 있는 객체 비교 |  결론  * **참조 동일성** 판단: == 연산자 또는 기본 equals() 사용[1](https://kephilab.tistory.com/92)[5](https://developer-yeony.tistory.com/107). * **내용 동일성** 판단: equals()를 재정의하고 hashCode()도 함께 구현[1](https://kephilab.tistory.com/92)[4](https://www.baeldung.com/java-comparing-objects). * **Null-safe 비교**: Objects.equals() 활용[4](https://www.baeldung.com/java-comparing-objects). |

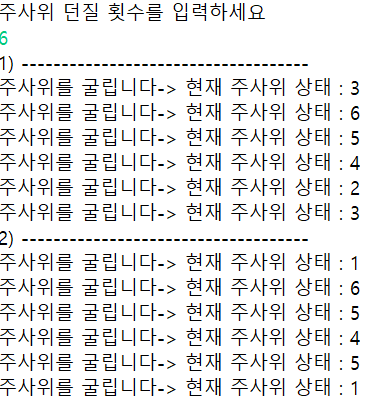
1. **[생성형 AI 활용]** 문자열처리와 관련하여 사용 빈도가 높은 메소드의 기능과 사용 예를 제시하세요

|  |  |
| --- | --- |
| 생성형 AI | perplexity |
| 질문 | 문자열처리와 관련하여 사용 빈도가 높은 메소드의 기능과 사용 예를 제시하세요 |
| 답변 | 문자열 처리에서 사용 빈도가 높은 메소드와 기능, 사용 예시 아래는 Java에서 문자열 처리 시 자주 사용되는 주요 메소드와 그 기능, 그리고 간단한 사용 예시입니다.  **1. length()**   * 기능: 문자열의 길이(문자 개수)를 반환합니다. * 예시:   java  String str = "Hello Java!";  System.out.println(str.length()); *// 11*  **2. charAt(int index)**   * 기능: 지정한 인덱스 위치의 문자를 반환합니다. * 예시:   java  String str = "Hello";  System.out.println(str.charAt(1)); *// 'e'*  **3. indexOf(String s) / lastIndexOf(String s)**   * 기능: 지정한 문자 또는 문자열이 처음(또는 마지막)으로 등장하는 인덱스를 반환합니다. 없으면 -1 반환. * 예시:   java  String str = "Hello Java!";  System.out.println(str.indexOf('a')); *// 7*  System.out.println(str.lastIndexOf('a')); *// 9*  **4. substring(int beginIndex, int endIndex)**   * 기능: 지정한 범위의 부분 문자열을 반환합니다. (endIndex는 포함하지 않음) * 예시:   java  String str = "Hello Java!";  System.out.println(str.substring(6, 10)); *// "Java"*  **5. equals(String anotherString) / equalsIgnoreCase(String anotherString)**   * 기능: 두 문자열의 내용이 같은지 비교합니다. equalsIgnoreCase는 대소문자 구분 없이 비교합니다. * 예시:   java  String str1 = "Java";  String str2 = "java";  System.out.println(str1.equals(str2)); *// false*  System.out.println(str1.equalsIgnoreCase(str2)); *// true*  **6. toLowerCase() / toUpperCase()**   * 기능: 영문자를 모두 소문자(또는 대문자)로 변환합니다. * 예시:   java  String str = "Java Study";  System.out.println(str.toLowerCase()); *// "java study"*  System.out.println(str.toUpperCase()); *// "JAVA STUDY"*  **7. replace(String target, String replacement)**   * 기능: 특정 문자열을 다른 문자열로 대체합니다. * 예시:   java  String str = "Java Study";  System.out.println(str.replace("Study", "공부")); *// "Java 공부"*  **8. split(String regex)**   * 기능: 지정한 구분자(정규표현식 기준)로 문자열을 분리하여 배열로 반환합니다. * 예시:   java  String str = "abc/def-ghi jkl";  String[] arr = str.split("/|-| ");  System.out.println(Arrays.toString(arr)); *// [abc, def, ghi, jkl]*  **9. trim()**   * 기능: 문자열 양쪽 끝의 공백을 제거합니다. * 예시:   java  String str = " Hello ";  System.out.println(str.trim()); *// "Hello"*  **10. isEmpty() / isBlank()**   * 기능: 문자열이 비어 있는지(길이 0) 또는 공백만 있는지 확인합니다. (isBlank는 Java 11 이상) * 예시:   java  String str = "";  System.out.println(str.isEmpty()); *// true*  **11. toCharArray()**   * 기능: 문자열을 char 배열로 변환합니다. * 예시:   java  String str = "Hello";  **char**[] chars = str.toCharArray();  System.out.println(Arrays.toString(chars)); *// [H, e, l, l, o]*  **12. valueOf(기본 자료형)**   * 기능: 기본 자료형을 문자열로 변환합니다. * 예시:   java  **int** num = 123;  String str = String.valueOf(num); *// "123"*  이 외에도 contains(), startsWith(), endsWith(), format(), repeat() 등 다양한 메소드가 자주 활용됩니다. 위 메소드들은 문자열 처리에서 매우 빈번하게 사용되며, 실제 프로젝트와 코딩테스트 등 다양한 상황에서 필수적으로 익혀두어야 합니다. |

(3) 2번에서 제시한 메소드를 사용하여 문자열을 처리하는 프로그램을 작성하세요. 단, 수업시간에 언급하지 않은 메소드를 사용해야 하며 사용 메소드는 4가지 이상이어야 합니다

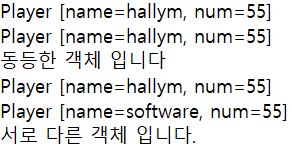
|  |
| --- |
| [프로그램 소스]  package practice2;  import java.sql.SQLOutput; import java.util.Arrays;  public class Main1 {  public static void main(String[] args) {   // lastIndexOf()  String str1 = "Hello World!";  System.*out*.println("str1 = " + str1);  System.*out*.println("lastIndexOf() = " + str1.lastIndexOf("l"));   // equalsIgnoreCase()  String str2 = "hello world!";  System.*out*.println("str2 = " + str2);  System.*out*.println("equalsIgnoreCase() = " + str1.equalsIgnoreCase(str2));   // isBlank()  String str3 = "";  System.*out*.println("str3 = " + str3);  System.*out*.println("isBlank() = " + str3.isBlank());   // toCharArray()  char[] chars = str1.toCharArray();  System.*out*.println("toCharArray() = " + Arrays.*toString*(chars));  } } |
| [실행 결과] |

(4) 입력 받은 값만큼 주사위를 던져서 나타나는 면의 숫자를 출력하는 프로그램을 작성하세요. 난수를 사용하며 난수 생성은 두가지 방법을 각각 사용하여야 합니다



|  |
| --- |
| [프로그램 소스]  package practice4;  import java.util.Random; import java.util.Scanner;  public class Main {  public static void main(String[] args) {  Scanner sc = new Scanner(System.*in*);  System.*out*.println("주사위 던질 횟수를 입력하세요");   int n = sc.nextInt();  System.*out*.println("1) --------------------------------------");  for (int i = 0; i < n; i++) {  int random1 = (int)(Math.*random*() \* 6) +1;  System.*out*.println("주사위를 굴립니다 -> 현재 주사위 상태 : " + random1);  }   System.*out*.println("2) --------------------------------------");  for (int i = 0; i < n; i++) {  Random random = new Random();  int random2 = random.nextInt(6) +1;  System.*out*.println("주사위를 굴립니다 -> 현재 주사위 상태 : " + random2);  }  } } |
| [실행 결과]  텍스트, 스크린샷, 소프트웨어, 번호이(가) 표시된 사진  AI가 생성한 콘텐츠는 부정확할 수 있습니다. |

(5) 다음과 같은 객체의 내용 비교를 할 수 있도록 Player 클래스를 작성하고 테스트하세요. equals()와 hashCode() 메소드를 모두 사용합니다.



class Player{

private String name;

private int num;

public Player(String name, int num) {

this.name = name;

this.num = num;

}

@Override

public boolean equals(Object obj) {

}

@Override

public int hashCode() {

}

@Override

public String toString() {

return "Player [name=" + name + ", num=" + num + "]";

}

}

|  |
| --- |
| [프로그램 소스]  package practice5;  import java.util.Objects;  class Player{  private String name;  private int num;   public Player(String name, int num) {  this.name = name;  this.num = num;  }   @Override  public boolean equals(Object obj) {  if(obj instanceof Player){  Player p = (Player) obj;  if(name.equals(p.name) && num == p.num){  return true;  }  }  return false;  }   @Override  public int hashCode() {  // name, age가 같으면 항상 같은 해시코드를 반환합니다.  return Objects.*hash*(name, num);  }   @Override  public String toString() {  return "Player [name=" + name + ", num=" + num + "]";  } }  package practice5;  public class Main {  public static void main(String[] args) {  Player player1 = new Player("hallym", 55);  Player player2 = new Player("hallym", 55);  Player player3 = new Player("software", 55);  System.*out*.println(player1);  System.*out*.println(player2);  if( player1.equals(player2) && (player1.hashCode() == player2.hashCode())) {  System.*out*.println("동등한 객체입니다.");  } else {  System.*out*.println("서로 다른 객체입니다.");  }   System.*out*.println(player1);  System.*out*.println(player3);  if( player1.equals(player3) && (player1.hashCode() == player3.hashCode())) {  System.*out*.println("동등한 객체입니다.");  } else {  System.*out*.println("서로 다른 객체입니다.");  }  } } |
| [실행 결과]  텍스트, 스크린샷, 폰트, 소프트웨어이(가) 표시된 사진  AI가 생성한 콘텐츠는 부정확할 수 있습니다. |

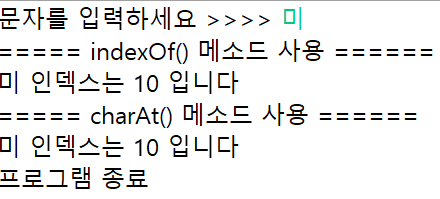
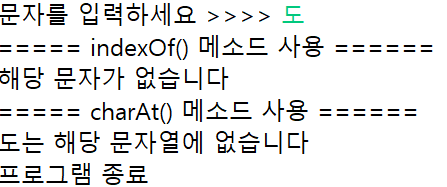
* **응용 프로그래밍**

(6) 두 자릿 수 로또 번호 6개를 생성하는 프로그램을 작성하세요. 단, 로또 번호는 중복될 수 없습니다. Random 클래스를 사용하여 난수를 생성합니다.

|  |
| --- |
| [프로그램 소스]  package practice6;  import java.util.Random;  public class Main {  public static void main(String[] args) {  int[] result = new int[6];  Random random = new Random();  int count = 0; // 배열 원소 개수 초기값 0   while(count < 6) { // count가 6까지 반복  int number = random.nextInt(45) + 1; // 1~45까지의 정수 난수 발생  boolean isDuplicate = false; // 중복유무 초기값 false   for (int i = 0; i < count; i++) {  // 중복체크  if(result[i] == number) {  isDuplicate = true;  break;  }  }   if(!isDuplicate) {  result[count] = number;  count++;  }  }  for(int i = 0; i < result.length; i++) {  System.*out*.print(result[i] + " ");  }  } } |
| [실행 결과]  텍스트, 스크린샷, 폰트, 번호이(가) 표시된 사진  AI가 생성한 콘텐츠는 부정확할 수 있습니다. |

(7) 다음과 같이 초기화된 문자열에서 표준 입력 장치로 입력된 문자가 몇 번째 있는지 위치 값을 출력하는 프로그램을 제시된 조건대로 작성하세요.

* String str= “자바 프로그램은 재미 없습니다. 어렵습니다”
* 조건1 : indexOf() 메소드 사용
* 조건2 : charAt() 메소드 사용

|  |
| --- |
| [프로그램 소스]  package practice7;  import java.util.Scanner;  public class Main {  public static void main(String[] args) {  Scanner sc = new Scanner(System.*in*);  String s = sc.next();  String str = "자바 프로그램은 재미 없습니다. 어렵습니다";   System.*out*.println("=====indexOf() 메소드 사용=====");  if(str.indexOf(s) == -1){  System.*out*.println("해당 문자가 없습니다.");  } else {  System.*out*.println(s + " 인덱스는 " + str.indexOf(s) + " 입니다.");  }   System.*out*.println("=====charAt() 메소드 사용=====");  boolean flag = false;  for(int i = 0; i < str.length(); i++) {  char ch = s.charAt(0);  if(ch == str.charAt(i)){  System.*out*.println(s + " 인덱스는 " + i + " 입니다.");  flag = true;  }  }   if(!flag){  System.*out*.println("해당 문자가 없습니다.");  }   sc.close();  System.*out*.println("프로그램 종료");  } } |
| [실행 결과]  텍스트, 스크린샷, 폰트, 소프트웨어이(가) 표시된 사진  AI가 생성한 콘텐츠는 부정확할 수 있습니다. 텍스트, 스크린샷, 폰트, 소프트웨어이(가) 표시된 사진  AI가 생성한 콘텐츠는 부정확할 수 있습니다. |

(8) 다음과 같은 문자열에서 첫번째 ‘a’와 두번째 ‘a’ 사이의 문자열을 추출하는 프로그램을 작성하세요

힌트) indexOf(str, index); =>index 부터 str 검색

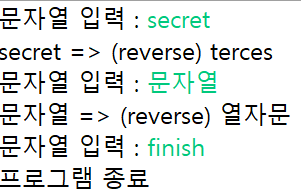
텍스트, 폰트, 라인, 스크린샷이(가) 표시된 사진

AI가 생성한 콘텐츠는 부정확할 수 있습니다.String str="hallymbigdata";

|  |
| --- |
| [프로그램 소스]  package practice8;  public class Main {  public static void main(String[] args) {  String str = "hallymbigdata";  int firstIdx = str.indexOf("a");  int secondIdx = str.indexOf("a", firstIdx+1);  String substr = str.substring(firstIdx+1, secondIdx);  System.*out*.println("str = " + str);  System.*out*.println("문자 a에서 a까지의 문자는 모두 " + substr.length() + "개 입니다.");  System.*out*.println("substr = " + substr);  } } |
| [실행 결과]  텍스트, 스크린샷, 폰트, 소프트웨어이(가) 표시된 사진  AI가 생성한 콘텐츠는 부정확할 수 있습니다. |

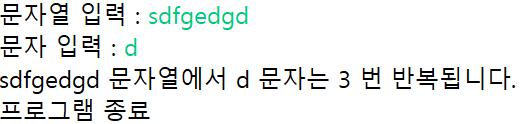
(9)입력 장치로 입력 받은 문자열을 역순으로 출력하는 프로그램을 제시된 조건대로 작성하세요. 예를 들어 사용자가 "secret"를 입력하면 "terces"를 출력합니다. 단, “finish”가 입력되면 반복을 종료합니다.

* 조건 1 : StringBuffer 클래스 reverse() 사용
* 조건 2 : charAt() 사용



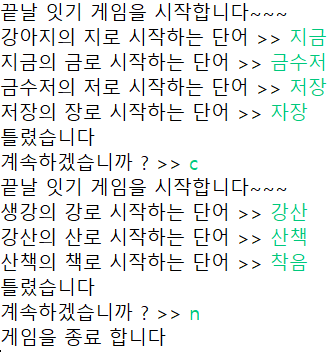
|  |
| --- |
| [프로그램 소스]  package practice9;  import java.util.Scanner;  public class Main {  public static void main(String[] args) {  Scanner sc = new Scanner(System.*in*);  while(true){  System.*out*.print("문자열 입력 : ");  String s = sc.next();   if(s.equals("finish")){  break;  }   // reverse  StringBuffer sb = new StringBuffer(s);  String reversedStr = sb.reverse().toString();  System.*out*.println(s + " => (reverse)" + reversedStr);  }  sc.close();  System.*out*.println("프로그램 종료");  } } |
| [실행 결과]  텍스트, 스크린샷, 폰트, 소프트웨어이(가) 표시된 사진  AI가 생성한 콘텐츠는 부정확할 수 있습니다. |

(10) 입력된 문자열에서 특정 문자가 반복되는 횟수를 출력하는 프로그램을 작성하세요. 단, indexOf()사용



|  |
| --- |
| [프로그램 소스]  package practice10;  import java.util.Scanner;  public class Main {  public static void main(String[] args) {  Scanner sc = new Scanner(System.*in*);   System.*out*.print("문자열 입력 : "); // sdfgedgd  String s = sc.next();   System.*out*.print("문자 입력 : ");  char ch = sc.next().charAt(0);   int count = 0;  for (int i = 0; i < s.length(); i++) {  if(s.charAt(i) == ch){  count++;  }  }  System.*out*.println(s + " 문자열에서 " + ch + " 문자는 " + count + "번 반복됩니다.");  sc.close();  System.*out*.println("프로그램 종료");  } } |
| [실행 결과] |

(11) 다음과 같이 끝말 잇기 프로그램을 작성하세요. 초기화 된 배열에서 랜덤하게 단어를 선택합니다. ‘c’ 또는 ‘C’가 입력되면 게임을 계속하고 그 외의 문자가 입력되면 종료합니다



String[] str1= {"오이", "강아지", "자바", "생강", "소나기", "우동", “언어” };

랜덤하게 1개 정해서

|  |
| --- |
| [프로그램 소스]  package practice11;  import java.util.Random; import java.util.Scanner;  public class Main {  public static void main(String[] args) {  Scanner sc = new Scanner(System.*in*);  String[] str = {"오이", "강아지", "자바", "생강", "소나기", "우동", "언어"};  Random random = new Random();   System.*out*.println("끝말 잇기 게임을 시작합니다~~~");  String randomStr = str[random.nextInt(str.length)];   while(true){  char randomLast = randomStr.charAt(randomStr.length()-1);  System.*out*.print(randomStr + "의 " + randomLast + "로 시작하는 단어 >> ");  String s = sc.next();   char first = s.charAt(0);   if(first == randomLast){  // 성공시 randomStr 갱신  randomStr = s;  } else {  // 실패시  System.*out*.println("틀렸습니다.");  System.*out*.print("계속하겠습니까 ? >> ");  s = sc.next();  if (s.equalsIgnoreCase("c")){  System.*out*.println("끝말 잇기 게임을 시작합니다~~~");  randomStr = str[random.nextInt(str.length)];  continue;  } else {  break;  }  }  }  sc.close();  System.*out*.println("게임을 종료 합니다.");  } } |
| [실행 결과]  텍스트, 스크린샷, 소프트웨어, 멀티미디어 소프트웨어이(가) 표시된 사진  AI가 생성한 콘텐츠는 부정확할 수 있습니다. |