Drew Sweeney

Dr. Hu

CSC 341

4/29/2022

Homework11 Reflection:

This homework wasn't too hard I feel that I was able to figure out the patterns and produce good outputs following the correct methods for each of the problems.

Package Chain:

Class Chain:

```
package chain;

import java.util.Random;

public class Chain
{
	Random rand = new Random();

	public Chain()
	{
		generateSentence();
	}

	//conjunction
	public String conjunction()
	{
		String[] conjunction = {"and", "or", "but", "because"};
		int random = rand.nextInt(conjunction.length);
		String word = conjunction[random];
		String sentence = "";
		sentence += word;
		return sentence + " ";
	}

	//properNoun
	public String properNoun()
	{
		String[] properNoun = {"Red", "Jane", "Richard Nixon", "Miss America"};
		int random = rand.nextInt(properNoun.length);
		String word = properNoun[random];
		String sentence = "";
		sentence += word;
		return sentence + " ";
	}

	//commonNoun
	public String commonNoun()
	{
		String[] commonNoun = {"man", "woman", "fish", "elephant", "unicorn"};
		int random = rand.nextInt(commonNoun.length);
		String word = commonNoun[random];
		String sentence = "";
		sentence += word;
		return sentence + " ";
	}

	//determiner
	public String determiner()
	{
		String[] determiner = {"a", "the", "every", "some"};
		int random = rand.nextInt(determiner.length);
```

```
		String word = determiner[random];
		String sentence = "";
		sentence += word;
		return sentence + " ";
	}

	//adjective
	public String adjective()
	{
		String[] adjective = {"big", "tiny", "pretty", "bald"};
		int random = rand.nextInt(adjective.length);
		String word = adjective[random];
		String sentence = "";
		sentence += word;
		return sentence + " ";
	}

	//Intransitive verb
	public String intransitiveVerb()
	{
		String[] intransitiveVerb = {"runs", "jumps", "talks", "sleeps"};
		int random = rand.nextInt(intransitiveVerb.length);
		String word = intransitiveVerb[random];
		String sentence = "";
		sentence += word;
		return sentence + " ";
	}

	//transitive verb
	public String transitiveVerb()
	{
		String[] transitiveVerb = {"loves", "hates", "sees", "knows", "looks
for", "finds"};
		int random = rand.nextInt(transitiveVerb.length);
		String word = transitiveVerb[random];
		String sentence = "";
		sentence += word;
		return sentence + " ";
	}

	//generate sentence
	public void generateSentence()
	{
		String sentence = "";

		sentence += properNoun() + transitiveVerb() + determiner()
		+ adjective() + commonNoun() + conjunction() + determiner()
		+ commonNoun() + transitiveVerb() + determiner() + adjective() +
commonNoun();

		System.out.println(sentence);
	}

}
```

Class Main:

```
package chain;

public class Main {

	public static void main(String[] args)
	{
		new Chain();
	}

}
```

Package observer:
Class Ba:

```
package observer;

public class Ba
{
	Message message = new Message();

	public boolean readOrIgnore()
	{
		String type = message.randomMessageType();
		String ba = "ba";

		if (ba == type)
		{
			return true;
		}
		else
		{
			return false;
		}
	}
}
```

Class Determine:

```
package observer;

public class Determine
{
	Dev dev = new Dev();
	Tt tt = new Tt();
	Ba ba = new Ba();
	Tl tl = new Tl();

	public Determine()
	{
		if (dev.readOrIgnore() == true)
		{
			System.out.println("Devs read message");
		}
		else if(tt.readOrIgnore() == true)
		{
			System.out.println("Tts read message");
		}
		else if(tl.readOrIgnore() == true)
		{
			System.out.println("Tls read message");
		}
		else if(ba.readOrIgnore() == true)
		{
			System.out.println("Bas read message");
		}
		else
		{
			System.out.println("All read message");
```

```
		}
	}

}
```

Class Dev:

```
package observer;

public class Dev
{
	Message message = new Message();

	public boolean readOrIgnore()
	{
		String type = message.randomMessageType();
		String dev = "dev";

		if (dev == type)
		{
			return true;
		}
		else
		{
			return false;
		}
	}
}
```

Class Message:

```
package observer;

import java.util.Random;

public class Message
{
	Random rand = new Random();

	public String randomMessageType()
	{
		String[] types = {"dev", "ba", "tl", "tt", "all"};
		int random = rand.nextInt(types.length);
		String word = types[random];

		return word;
	}
}
```

Class Tl:

```
package observer;

public class Tl
{
```

```
	Message message = new Message();

	public boolean readOrIgnore()
	{
		String type = message.randomMessageType();
		String tl = "tl";

		if (tl == type)
		{
			return true;
		}
		else
		{
			return false;
		}
	}
}
```

Class Tt:

```
package observer;

public class Tt
{
	Message message = new Message();

	public boolean readOrIgnore()
	{
		String type = message.randomMessageType();
		String tt = "tt";

		if (tt == type)
		{
			return true;
		}
		else
		{
			return false;
		}
	}
}
```

Class Main:

```
package observer;

public class Main {

	public static void main(String[] args)
	{
		new Determine();
	}

}
```

Package Visitor:
Class adjective:

```java
package visitor;

import java.util.Random;

public class Adjective implements Word
{
	Random rand = new Random();

	 //adjective
	 public String adjective()
	 {
		String[] adjective = {"big", "tiny", "pretty", "bald"};
		int random = rand.nextInt(adjective.length);
		String word = adjective[random];
		String sentence = "";
		sentence += word;
		return sentence + " ";
	 }

	@Override
	public void accept(WordVisitor v)
	{
		v.add(adjective());

	}
}
```

Class Characters:

```java
package visitor;

import java.util.Random;

public class Characters implements Word
{

	Random rand = new Random();

	 //characters
	 public String character()
	 {
		String[] character = {"#", "%", "~", "+"};
		int random = rand.nextInt(character.length);
		String word = character[random];
		String sentence = "";
		sentence += word;
		return sentence + " ";
	 }

	@Override
	public void accept(WordVisitor v)
	{
		v.add(character());
	}
```

```
}
```

Class commonNoun:

```
package visitor;

import java.util.Random;

public class CommonNoun implements Word
{
	Random rand = new Random();

	 //commonNoun
	 public String commonNoun()
	 {
		String[] commonNoun = {"man", "woman", "fish", "elephant", "unicorn"};
		int random = rand.nextInt(commonNoun.length);
		String word = commonNoun[random];
		String sentence = "";
		sentence += word;
		return sentence + " ";
	 }

	 @Override
		public void accept(WordVisitor v)
		{
			v.add(commonNoun());
		}
}
```

Class Conjunction:

```
package visitor;

import java.util.Random;

public class Conjunction implements Word
{
	Random rand = new Random();

	 //conjunction
	 public String conjunction()
	 {
		String[] conjunction = {"and", "or", "but", "because"};
		int random = rand.nextInt(conjunction.length);
		String word = conjunction[random];
		String sentence = "";
		sentence += word;
		return sentence + " ";
	 }

	 @Override
	public void accept(WordVisitor v)
	{
		v.add(conjunction());
	}
```

```
}
```

Class Determiner:

```
package visitor;

import java.util.Random;

public class Determiner implements Word
{
	Random rand = new Random();

	 //determiner
	 public String determiner()
	 {
		String[] determiner = {"a", "the", "every", "some"};
		int random = rand.nextInt(determiner.length);
		String word = determiner[random];
		String sentence = "";
		sentence += word;
		return sentence + " ";
	 }

	 @Override
	public void accept(WordVisitor v)
	{
		v.add(determiner());

	}

}
```

Class IntransitiveVerb:

```
package visitor;

import java.util.Random;

public class IntransitiveVerb implements Word
{
	Random rand = new Random();

	 //Intransitive verb
	 public String intransitiveVerb()
	 {
		String[] intransitiveVerb = {"runs", "jumps", "talks", "sleeps"};
		int random = rand.nextInt(intransitiveVerb.length);
		String word = intransitiveVerb[random];
		String sentence = "";
		sentence += word;
		return sentence + " ";
	 }

	 @Override
		public void accept(WordVisitor v)
		{
```

```
                v.add(intransitiveVerb());
            }
}


Class ProperNoun:
package visitor;

import java.util.Random;

public class ProperNoun implements Word
{
	Random rand = new Random();

	public ProperNoun()
	{
		properNoun();
	}

	 //properNoun
	 public String properNoun()
	 {
		 String[] properNoun = {"Red", "Jane", "Richard Nixon", "Miss America"};
		 int random = rand.nextInt(properNoun.length);
		 String word = properNoun[random];
		 String sentence = "";
		 sentence += word;
		 return sentence + " ";
	 }

	 @Override
		public void accept(WordVisitor v)
		{
			v.add(properNoun());
		}
}

Class TransitiveVerb:
package visitor;

import java.util.Random;

public class TransitiveVerb implements Word
{
	Random rand = new Random();

	 //transitive verb
	 public String transitiveVerb()
	 {
		 String[] transitiveVerb = {"loves", "hates", "sees", "knows", "looks
for", "finds"};
		 int random = rand.nextInt(transitiveVerb.length);
		 String word = transitiveVerb[random];
		 String sentence = "";
		 sentence += word;
```

```
                return sentence + " ";
        }

        @Override
              public void accept(WordVisitor v)
              {
                    v.add(transitiveVerb());
              }
}
```

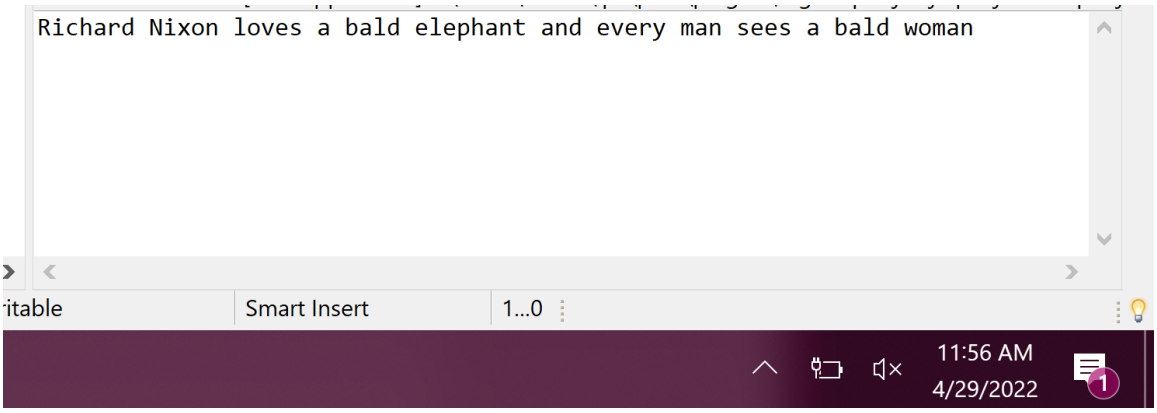
Interface Word:

```
package visitor;

public interface Word
{
      void accept(WordVisitor v);
}
```

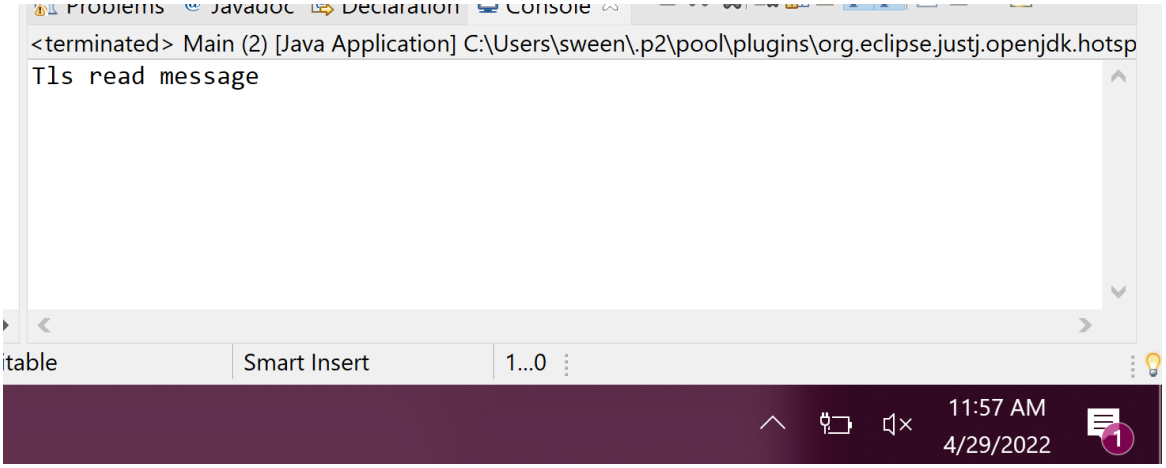
Class WordVisitor:

```
package visitor;

public class WordVisitor
{

      String sentence = "";

      public WordVisitor()
      {
            String sentence = "";
            generateSentence();
      }

      public String add(String word)
      {
            sentence += word;
            return sentence;
      }

      public void generateSentence()
      {
            System.out.println(sentence);
      }

}
```

Chain Output:




Observer Output:




Visitor Output: