Problem Description: To draw 4 monkeys side by side by making use of patterns.

Problem solution: Created a method for each character(such as \* or /)with repeating parameter.

And created methods to draw monkeys' one line or repeated parts. Used for some loops to draw those repeated parts.

Implementation:

public class OFO2016400048 {

public static void main(String []args) {

//line1

for(int i=0;i<4;i++) {

space(5);

top1();

space(5);

}

System.out.println();

//line2

space(3);

top2a();

space(3);

for(int i=0;i<3;i++) {

space(3);

top2b();

space(3);

}

System.out.println();

//line3

space(2);

slash(1);

space(1);

eyes1();

space(1);

bslash(1);

space(4);

slash(1);

stick(1);

eyes2();

stick(1);

bslash(1);

space(2);

for(int i=0;i<2;i++) {

space(2);

para(1);

space(1);

eyes2();

space(1);

para2(1);

space(2);

}

System.out.println();

//line4

space(1);

slash(1);

space(1);

mouth1a();

space(1);

bslash(1);

space(2);

stick(1);

space(1);

mouth1a();

space(1);

stick(1);

space(1);

for(int i=0;i<2;i++) {

space(3);

mouth1b();

space(3);

}

System.out.println();

//line5

for(int i=0;i<2;i++)

mouth2a();

space(4);

mouth2b();

space(8);

mouth2c();

System.out.println();

//line6

for(int i=0;i<2;i++)

neck1a();

space(4);

neck1b();

space(8);

neck1c();

System.out.println();

//line8

for(int i=0;i<2;i++) {

space(1);

neck2a();

space(1);

}

space(3);

neck2b();

space(6);

neck2c();

System.out.println();

//line9

System.out.print(" {see no evil} ");

System.out.print("{hear no evil} ");

System.out.print("{speak no evil} ");

System.out.print("{have no fun}");

}

public static void space(int a) {

for(int i=0;i<a;i++) {

System.out.print(" ");

}

}

//Draws top1

public static void top1() {

dot(1);

line(1);

nail(1);

line(1);

dot(1);

//System.out.print(".-\"-.");

}

//Draws top2a

public static void top2a() {

uline(1);

slash(1);

uline(1);

line(1);

dot(1);

line(1);

uline(1);

bslash(1);

uline(1);

//System.out.print("\_/\_-.-\_\\\_");

}

//Draws top2b

public static void top2b() {

uline(1);

slash(1);

dot(1);

line(1);

dot(1);

line(1);

dot(1);

bslash(1);

uline(1);

//System.out.print("\_/.-.-.\\\_");

}

//Draws eyes1

public static void eyes1() {

uline(2);

parb2(1);

space(1);

parb(1);

uline(2);

//System.out.print("\_\_} {\_\_");

}

//Draws eyes2

public static void eyes2() {

para(1);

space(1);

o(1);

space(1);

o(1);

space(1);

para2(1);

//System.out.print("( o o )");

}

//Draws mouth1a

public static void mouth1a() {

slash(2);

space(2);

nail(1);

space(2);

bslash(2);

//System.out.print("// \" \\\\");

}

//Draws mouth1b

public static void mouth1b() {

stick(1);

slash(1);

space(2);

nail(1);

space(2);

bslash(1);

stick(1);

//System.out.print("|/ \" [\\|](file://\\|\)");

}

//Draws mouth2a

public static void mouth2a() {

slash(1);

space(1);

slash(1);

space(1);

bslash(1);

bkes(1);

line(3);

bkes(1);

slash(1);

space(1);

bslash(1);

space(1);

bslash(1);

//System.out.print("/ / [\\’---’/](file://\\’---’\) \\ \\");

}

//Draws mouth2b

public static void mouth2b() {

bslash(1);

bkes(1);

slash(1);

hat(1);

bslash(1);

bkes(1);

slash(1);

//System.out.print("\\’/ˆ\\’/");

}

//Draws mouth2c

public static void mouth2c() {

bslash(1);

space(1);

dot(1);

line(1);

dot(1);

space(1);

slash(1);

//System.out.print("\\ .-. /");

}

//Draws neck1a

public static void neck1a() {

bslash(1);

space(1);

bslash(1);

uline(1);

neck1c();

uline(1);

slash(1);

space(1);

slash(1);

//System.out.print("\\ [\\\_/‘\"\"\"‘\\\_/](file://\\_\‘\%22\%22\%22‘\\_\) /");

}

//Draws neck1b

public static void neck1b() {

slash(1);

kes(1);

bslash(1);

space(1);

slash(1);

kes(1);

bslash(1);

//System.out.print("/‘\\ /‘\\");

}

//Draws neck1c

public static void neck1c() {

slash(1);

kes(1);

nail(3);

kes(1);

bslash(1);

//System.out.print("/‘\"\"\"‘\\");

}

//Draws neck2a

public static void neck2a() {

bslash(1);

space(11);

slash(1);

//System.out.print(" \\ / ");

}

//Draws neck2b

public static void neck2b() {

slash(1);

space(2);

slash(1);

stick(1);

bslash(1);

space(2);

bslash(1);

//System.out.print("/ /|\\ \\");

}

//Draws neck2c

public static void neck2c() {

slash(1);

space(7);

bslash(1);

//System.out.print("/ \\");

}

//Every method draws each character with repating parameter.

public static void slash(int a){

for(int i=0;i<a;i++)

System.out.print("/");

}

public static void bslash(int a){

for(int i=0;i<a;i++)

System.out.print("\\");

}

public static void star(int a){

for(int i=0;i<a;i++)

System.out.print("\*");

}

public static void nail(int a){

for(int i=0;i<a;i++)

System.out.print("\"");

}

public static void stick(int a){

for(int i=0;i<a;i++)

System.out.print("|");

}

public static void line(int a){

for(int i=0;i<a;i++)

System.out.print("-");

}

public static void uline(int a){

for(int i=0;i<a;i++)

System.out.print("\_");

}

public static void hat(int a){

for(int i=0;i<a;i++)

System.out.print("^");

}

public static void kes(int a){

for(int i=0;i<a;i++)

System.out.print("‘");

}

public static void bkes(int a){

for(int i=0;i<a;i++)

System.out.print("’");

}

public static void dot(int a){

for(int i=0;i<a;i++)

System.out.print(".");

}

public static void para(int a){

for(int i=0;i<a;i++)

System.out.print("(");

}

public static void para2(int a){

for(int i=0;i<a;i++)

System.out.print(")");

}

public static void parb(int a){

for(int i=0;i<a;i++)

System.out.print("{");

}

public static void parb2(int a){

for(int i=0;i<a;i++)

System.out.print("}");

}

public static void o(int a){

for(int i=0;i<a;i++)

System.out.print("o");

}

}



Conclusion: Picture generated correctly.

Due to the mailing with lab assistant i added methods with repeating factor for each character(such as \* or /) and i believe that created disorginazation in code. Old parts still stays in methods like //System.out.print("/‘\\ /‘\\");