

DSL - Linguagens de Domínio Específico

Trabalho final

Inácia Fernandes da Costa Neta
Philiply Augusto Silveira de Brito

Contexto

- Necessidade de facilitar o entendimento de cores e sua combinações (primárias, secundárias e terciárias) em inglês por crianças
- Linguagem lúdica para fácil compreensão
- Associação entre palavras, cores e idiomas (português e inglês)

Grammar

```

grammar Colors;

prog : entrada+
    ;

entrada : VAR '=' exp    #Igualdade
        | 'combinar' exp #Combinar
        | 'mostrar' exp  #Mostrar
        ;

exp : exp '+' exp #Combinacao
    | VAR         #Var
    | cores       #Cor
    ;

cores: 'azul'      #Azul
      | 'amarelo'  #Amarelo
      | 'vermelho' #Vermelho
      | 'laranja'  #Laranja
      | 'verde'    #Verde
      | 'roxo'     #Roxo
      | 'cinza'    #Cinza
      ;

VAR : [a-zA-Z]+
    ;

SPACE : [ \n\t\r] -> skip
    ;

```

Visitor

```
@Override
public String visitProg(ColorsParser.ProgContext ctx) {
    String preProg = "";
    preProg =
        "from PIL import Image, ImageFilter, ImageColor, ImageDraw, ImageFont\n" +
```

```
img = Image.new('RGB', (256, 256), ImageColor.getrgb(cor))\n" +
img.save(nomeCor+'.jpg', 'JPEG')\n" +
base = Image.open(nomeCor+'.jpg').convert('RGBA')\n" +
txt = Image.new('RGBA', base.size, (255,255,255,0))\n" +
fnt = ImageFont.truetype('stocky.ttf', 40)\n" +
d = ImageDraw.Draw(txt)\n"+
d.text((10,60), nomeCor, font=fnt, fill=(255,255,255,255))\n"+
out = Image.alpha_composite(base, txt)\n"+
out.save(nomeCor+'.png', 'PNG')\n"+
imagem = nomeCor + '.png'\n"+
im = Image.open(imagem)\n"+
im.show()\n"+
```

Visitor

```
@Override
public String visitIgualdade(ColorsParser.IgualdadeContext ctx){
    String sVar = ctx.VAR().getText();
    String sExp = visit(ctx.exp());
    return sVar + "=" + sExp + "\n";
}

@Override
public String visitCombinar(ColorsParser.CombinarContext ctx) {
    String sExp = visit(ctx.exp());
    String combinaCor =
        "try:\n"+
        "    combinaCor(" + sExp + ")\n";
    return combinaCor + erroComb;
}

@Override
public String visitMostrar(ColorsParser.MostrarContext ctx) {
    String sExp = visit(ctx.exp());
    String gerarImagem =
        "try:\n"+
        "    geraImagem(" + sExp + ")\n";
    return gerarImagem + erroMost;
}
```

Visitor

```
@Override
public String visitCor(ColorsParser.CorContext ctx) {
    String sCor = visit(ctx.cores());
    return sCor;
}

@Override
public String visitAzul(ColorsParser.AzulContext ctx) {
    String azul = "blue";
    return azul;
}

@Override
public String visitAmarelo(ColorsParser.AmareloContext ctx) {
    String amarelo = "yellow";
    return amarelo;
}
```

Visitor

```
@Override  
public String visitCombinacao(ColorsParser.CombinacaoContext ctx) {  
    corEsquerda = visit(ctx.exp(0));  
    corDireita = visit(ctx.exp(1));  
    return corEsquerda + "," + corDireita;  
}
```

Testes

teste - Bloco de Notas

Arquivo Editar Formatar Exibir Ajuda

```
a = azul  
mostrar a
```

teste3 - Bloco de Notas

Arquivo Editar Formatar Exibir Ajuda

```
a = marrom  
mostrar a
```

teste4 - Bloco de Notas

Arquivo Editar Formatar Exibir Ajuda

```
a = azul  
b = verde  
combinar a + b
```

teste6 - Bloco de Notas

Arquivo Editar Formatar Exibir Ajuda

```
a = amarelo  
combinar a + b
```

teste7 - Bloco de Notas

Arquivo Editar Formatar Exibir Ajuda

```
a = vermelho  
b = cinza  
combinar a + b
```


Retorno

Em caso de apresentação de cor simples



Em caso de combinação das cores:

amarelo + laranja



Retorno

Em caso de apresentação de cor simples mas que não pode ser combinada com outras cores



Retorno

Em caso de erro

```
C:\antlr4\sample\trab2_final>java CorVisitor.java teste6.exp > teste.py
```

```
C:\antlr4\sample\trab2_final>py teste.py  
Nao foi possivel combinar as cores. Verifique as cores digitadas
```

```
C:\antlr4\sample\trab2_final>java CorVisitor.java teste7.exp > teste.py
```

```
C:\antlr4\sample\trab2_final>py teste.py  
Nao foi possivel combinar as cores red e grey
```

Obrigado!