

# MEDIEVAL DSL?

PARSING HERALDIC BLAZON WITH PYTHON

---

Lady Red aka Chris Beacham (She)

# COAT OF ARMS OF THE UK

Retrieved From Wikipedia  
Sodacan [Cc By-Sa 3.0]



CREST





DIEU ET

MON DROIT







SUPPORTERS

SUPPORTERS

DIEU ET

MON DROIT









MOTTO

DIEU ET

MON DROIT

## BLAZON FOR UK

---

- ▶ Quarterly, First and Fourth Gules three lions passant guardant in pale Or armed and langued Azure (for England), Second quarter Or a lion rampant within a double tressure flory counter-flory Gules (for Scotland), Third quarter Azure a harp Or stringed Argent (for Ireland), the whole surrounded by the Garter; for a Crest, upon the Royal helm the imperial crown Proper, thereon a lion statant guardant Or imperially crowned Proper; Mantling Or and ermine; for Supporters, dexter a lion rampant guardant Or crowned as the Crest, sinister a unicorn Argent armed, crined and unguled Proper, gorged with a coronet Or composed of crosses patée and fleurs de lys a chain affixed thereto passing between the forelegs and reflexed over the back also Or; Motto 'Dieu et mon Droit' in the compartment below the shield, with the Union rose, shamrock and thistle engrafted on the same stem.

**WE WILL PARSE THAT!**

- 
- ▶ Arms are heritable the way land is, not the way a name is.  
Don't get tricked by scammy websites! You likely have not inherited arms unless you also have a castle.
  - ▶ Rules vary from country to country and across time
  - ▶ Blazon is a remarkably consistent and rule-based language for a practice done over 500 years in the High Middle Ages by groups constantly at war with each other.
  - ▶ I have SIMPLIFIED somewhat (a lot)

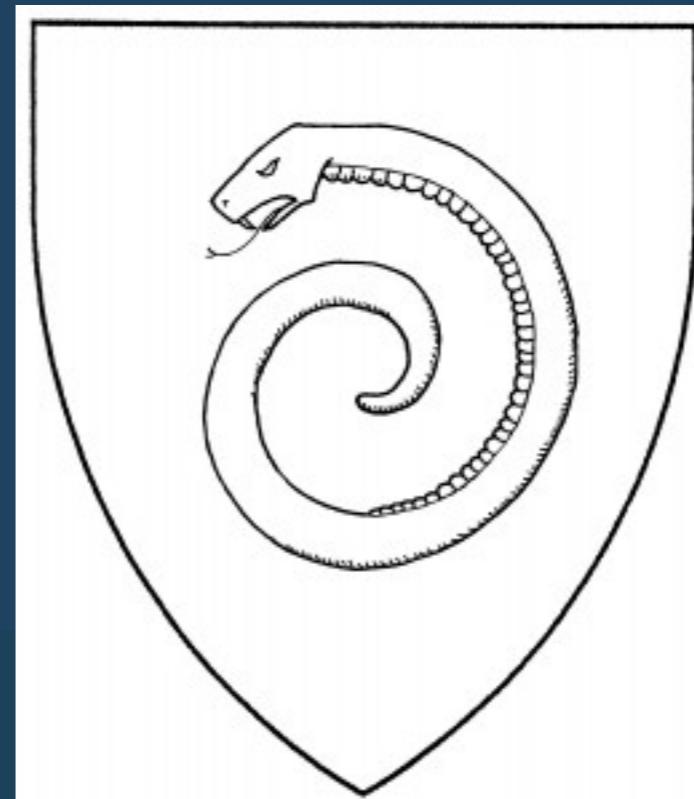
*Azure,*



*Azure, Two serpents erect addorsed  
entwined,*



PITHON



SERPENT

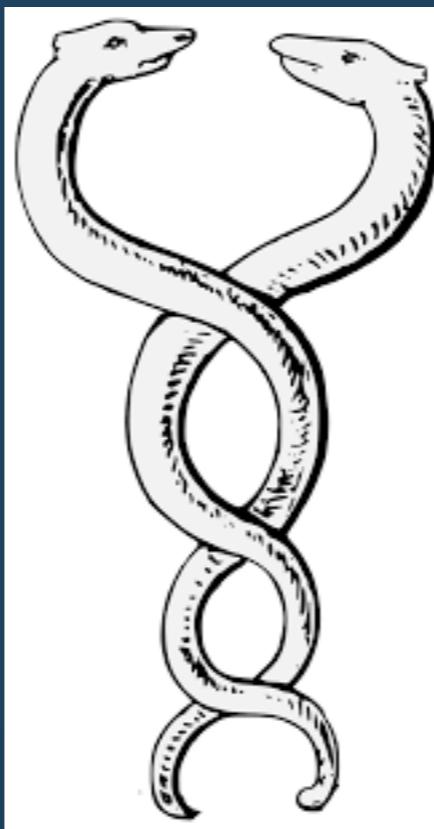


*Azure, Two serpents erect addorsed  
entwined,*



Source: dragon\_azure.tripod.com

*Azure, Two serpents erect addorsed  
entwined,*



*Azure, Two serpents erect addorsed entwined,  
of the field and Or, the sinister inverted*





LET'S START PARSING!

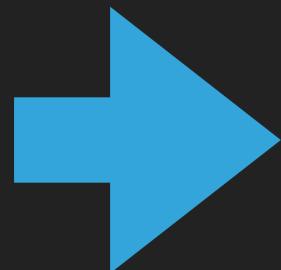
- ▶ We'll start trying to build a parser for Blazon
- ▶ Plenty of python parsers -
  - ▶ ANTLR / Lrparsing / PLY / Pyleri ... MANY more!
- ▶ Lark is our parser library ( [github.com/lark-parser/lark](https://github.com/lark-parser/lark) )
  - ▶ Easy interface
  - ▶ Several parser implementations with different characteristics
  - ▶ Rules are in Backus-Naur form (BNF), which is pretty standard
  - ▶ Supports left recursive rules
  - ▶ Good documentation

```
from lark import Lark

l = Lark('''

LCASE_LETTER: "a".."z"
UCASE_LETTER: "A".."Z"
LETTER: UCASE_LETTER | LCASE_LETTER
WORD: LETTER+
start: WORD "," WORD "!"
%ignore " "           // Disregard spaces in text
''')

print( l.parse("Hello, World!") )
```



```
Tree(start, [Token(WORD, 'Hello'), Token(WORD, 'World')])
```

## PARSING TINCTURES

<b><i>Metals</i></b>
<i>Or</i>
<i>Argent</i>
<b><i>Colours</i></b>
<i>Azure</i>
<i>Gules</i>
<i>Purpure</i>
<i>Sable</i>
<i>Vert</i>

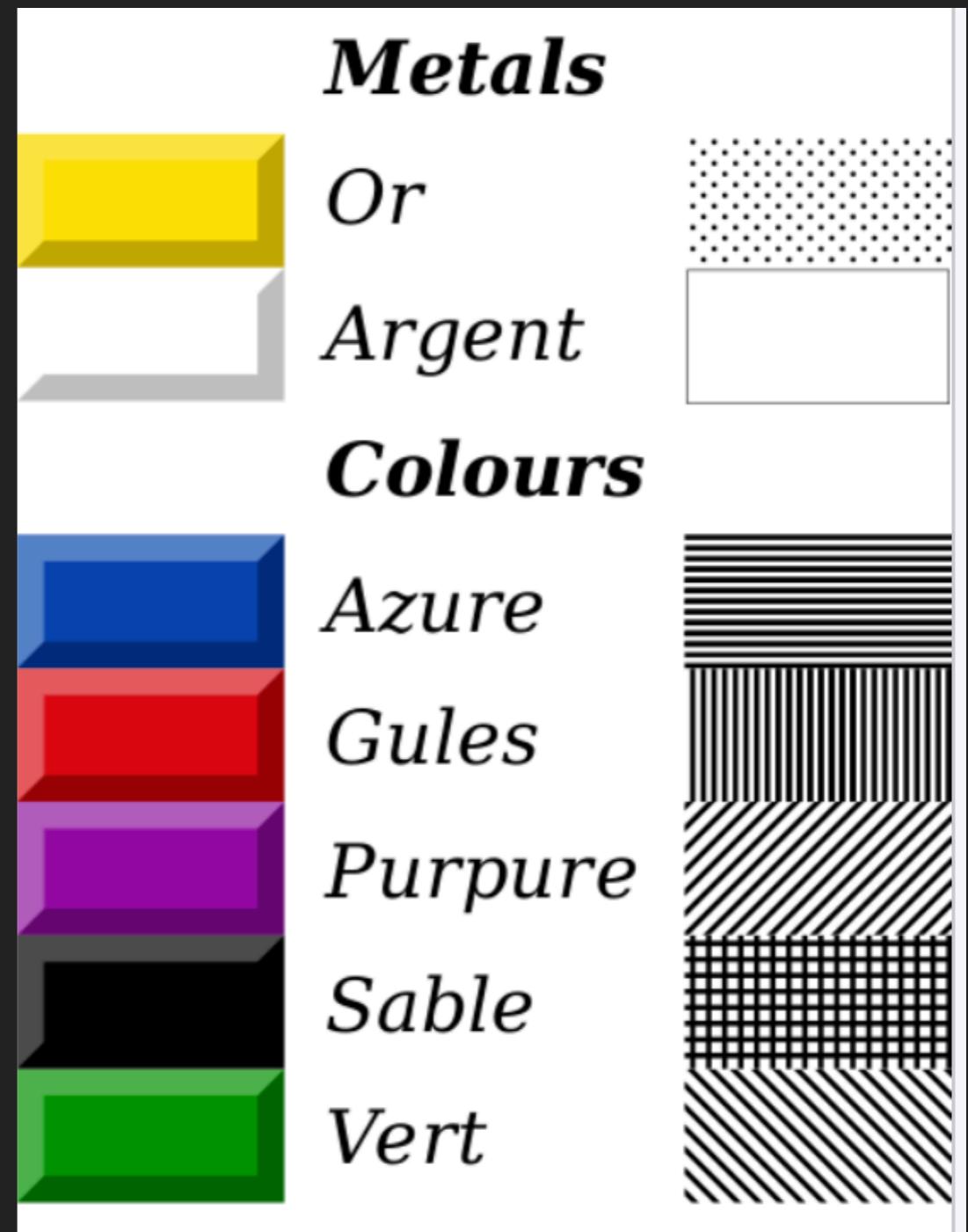


## PARSING TINCTURES

```
COLOUR: "Azure"i      // Blue
| "Gules"i        // Red
| "Purpure"i     // Purple
| "Sable"i        // Black
| "Vert"i         // Green
| "Proper"i       // Natural

METAL: "Or"i          // Gooooold
| "Argent"i      // Silver or White

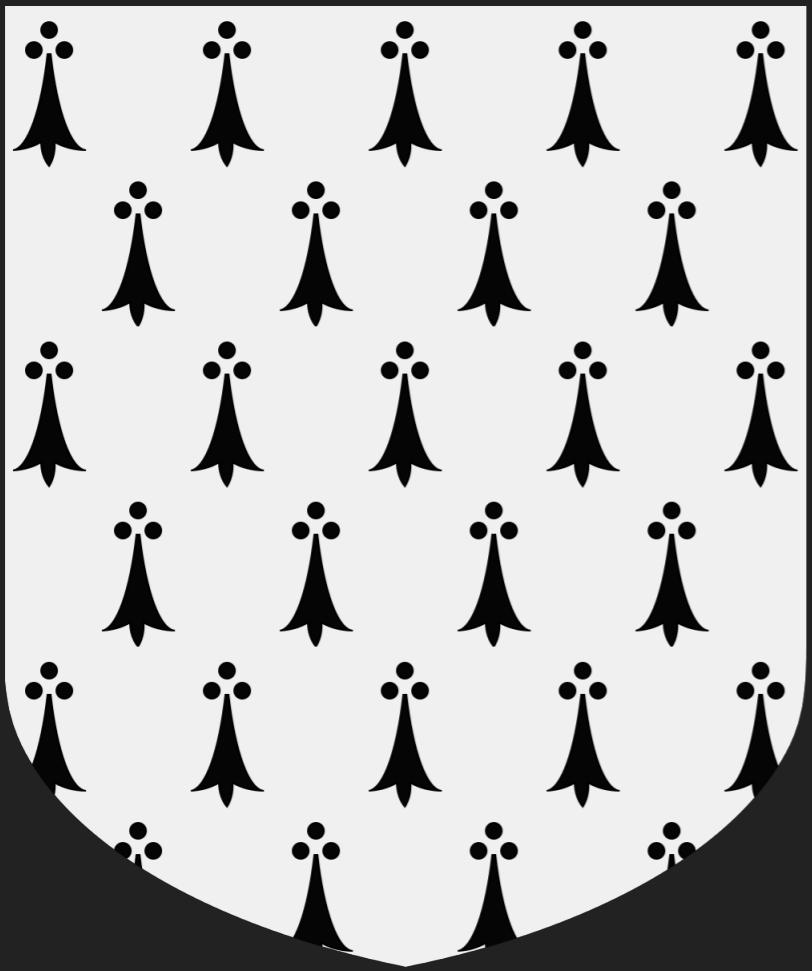
tincture: METAL | COLOUR
```



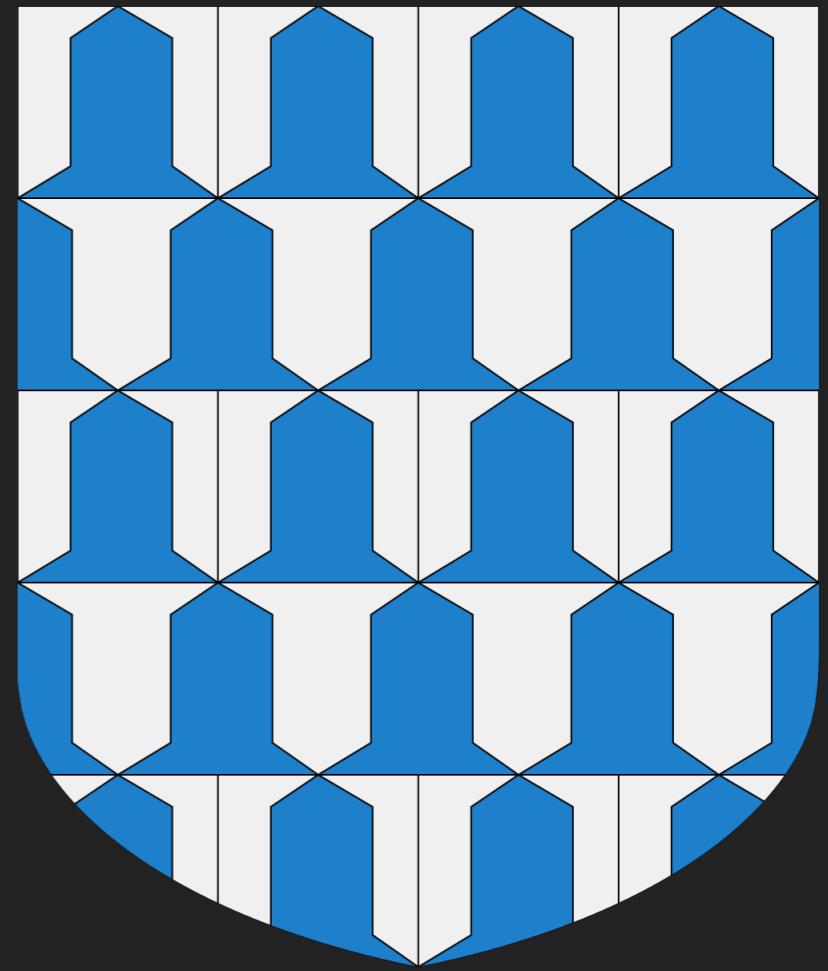


Azure

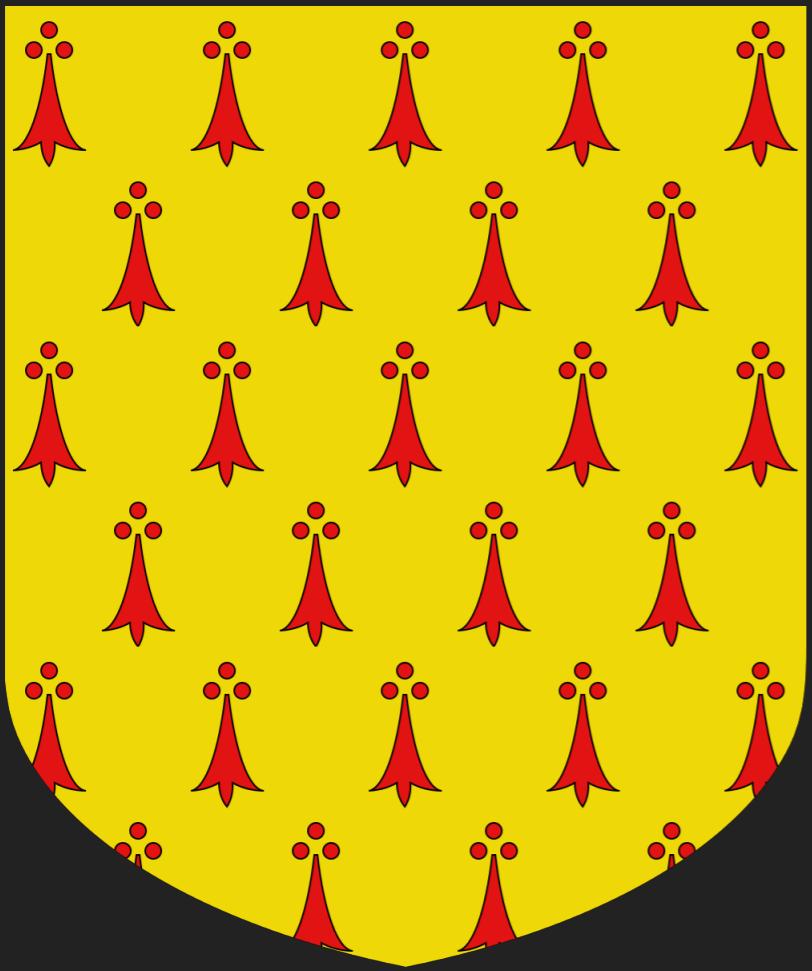
Or



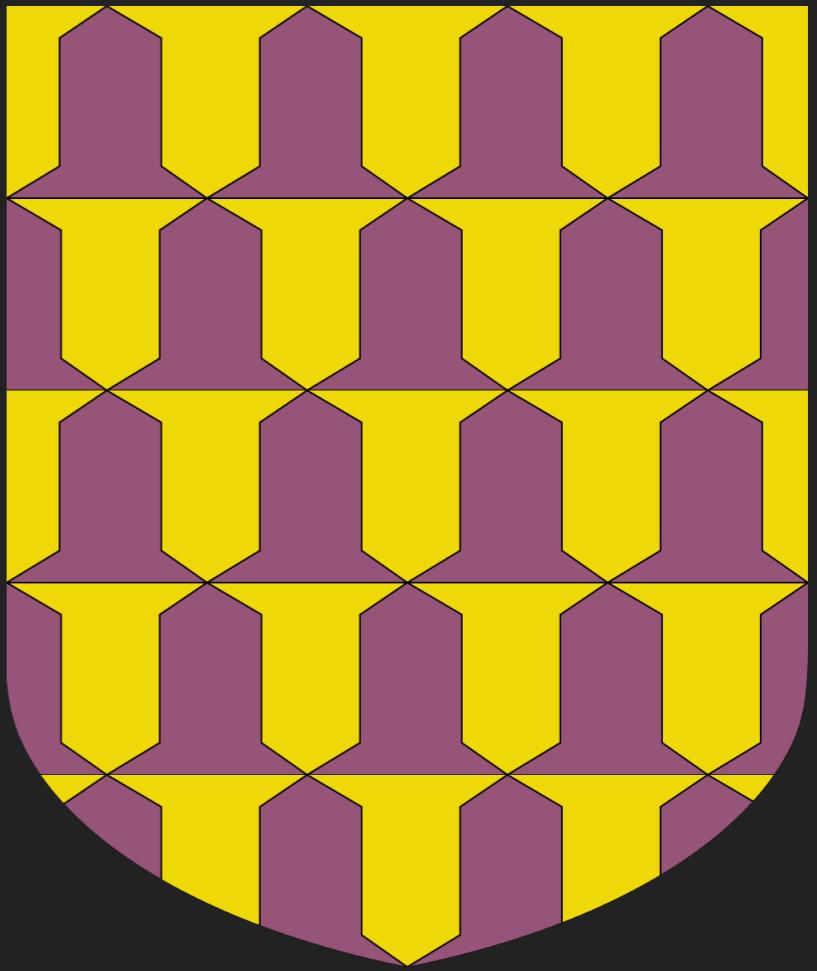
*Ermine*



*Vair*



*Or ermined Gules*



*Vairy Or and Purpure*



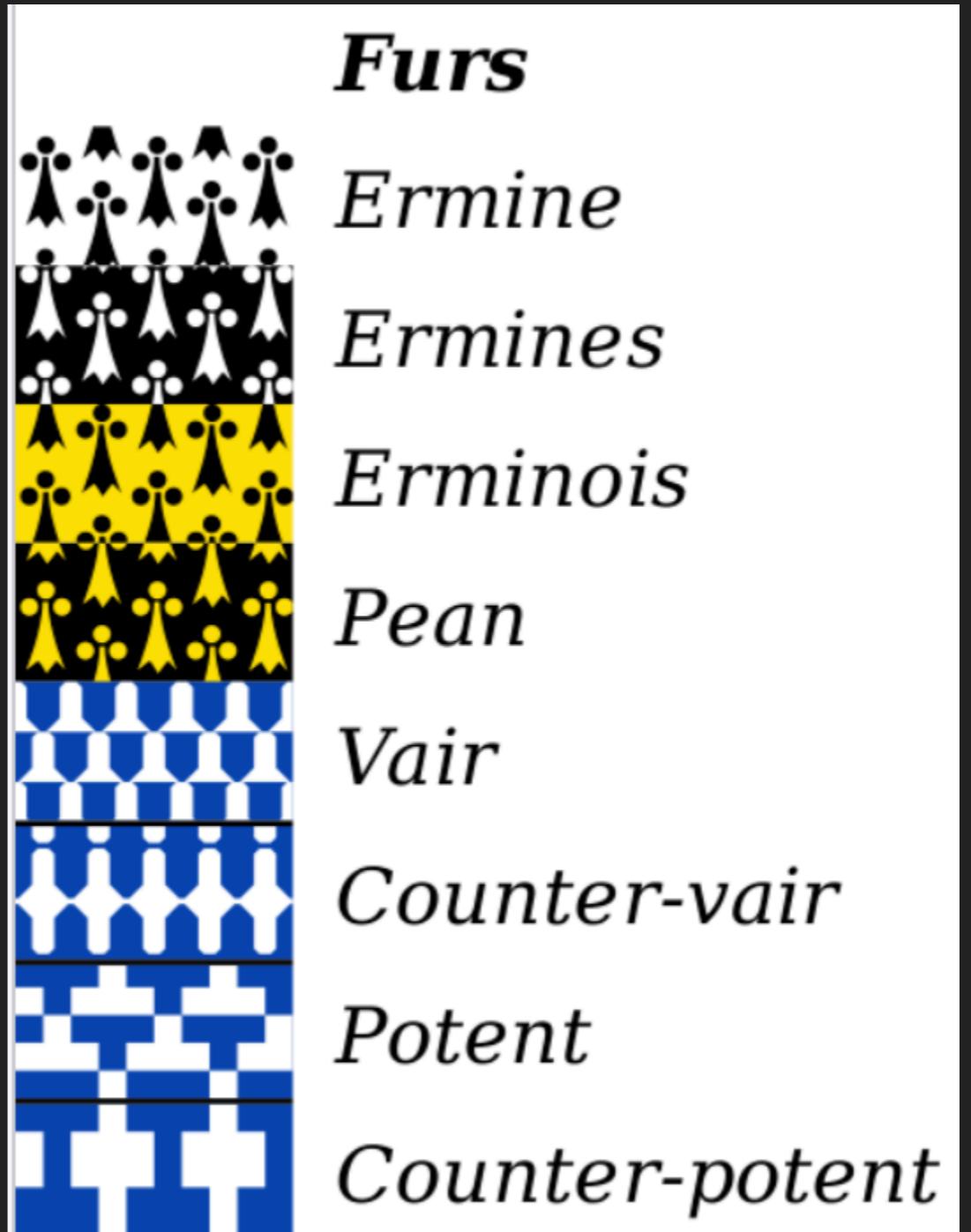
```
fur: ERMINE
| VAIR
| fur tincture AND tincture
| tincture INFIX_FUR tincture

// Stoat Fur design (black and white spots).
ERMINE: "Ermine"i
| "Ermines"i
| "Erminois"i
| "Pean"i

VAIR: "Vairy"i | "Vair"i // Bell-shaped pattern
| "Potent"i // Tetris T's alternating
| "Counter-Potent"i // Pluses alternating

INFIX_FUR: "Ermined"i
AND: "And"i

tincture: METAL | COLOUR | fur
```



---

## BASE CASES FIELDS

start: arms

arms: field

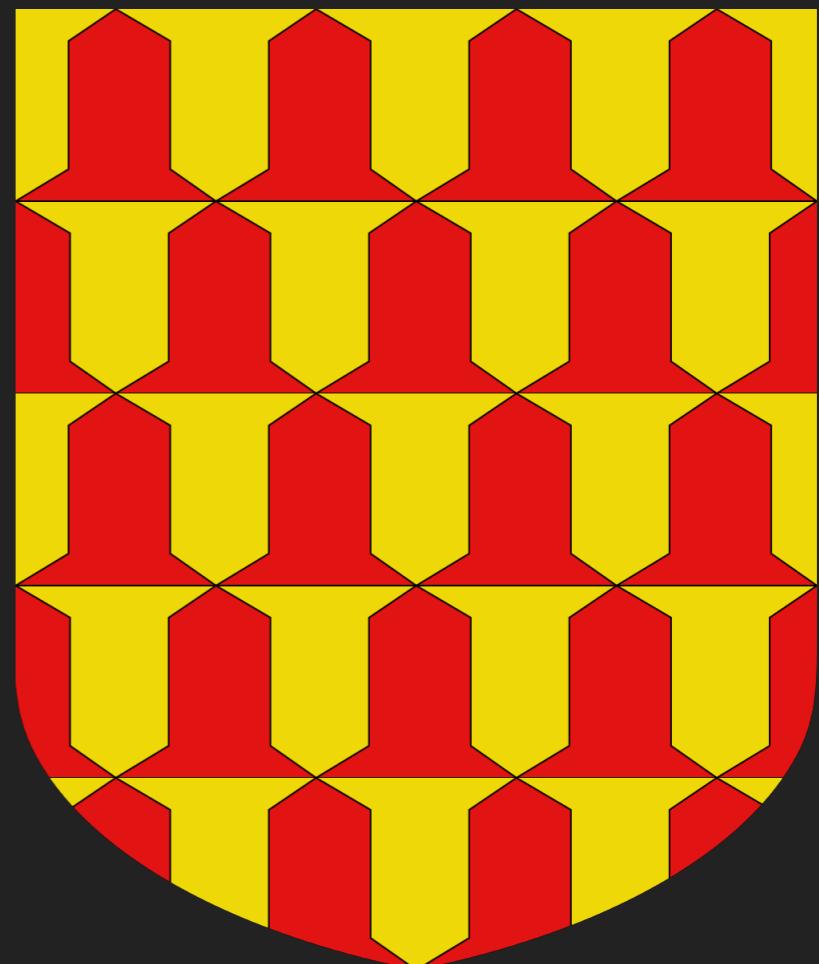
field: tincture

(these will get more complex)

## HERALDIC ABSTRACT SYNTAX TREE (HAST)

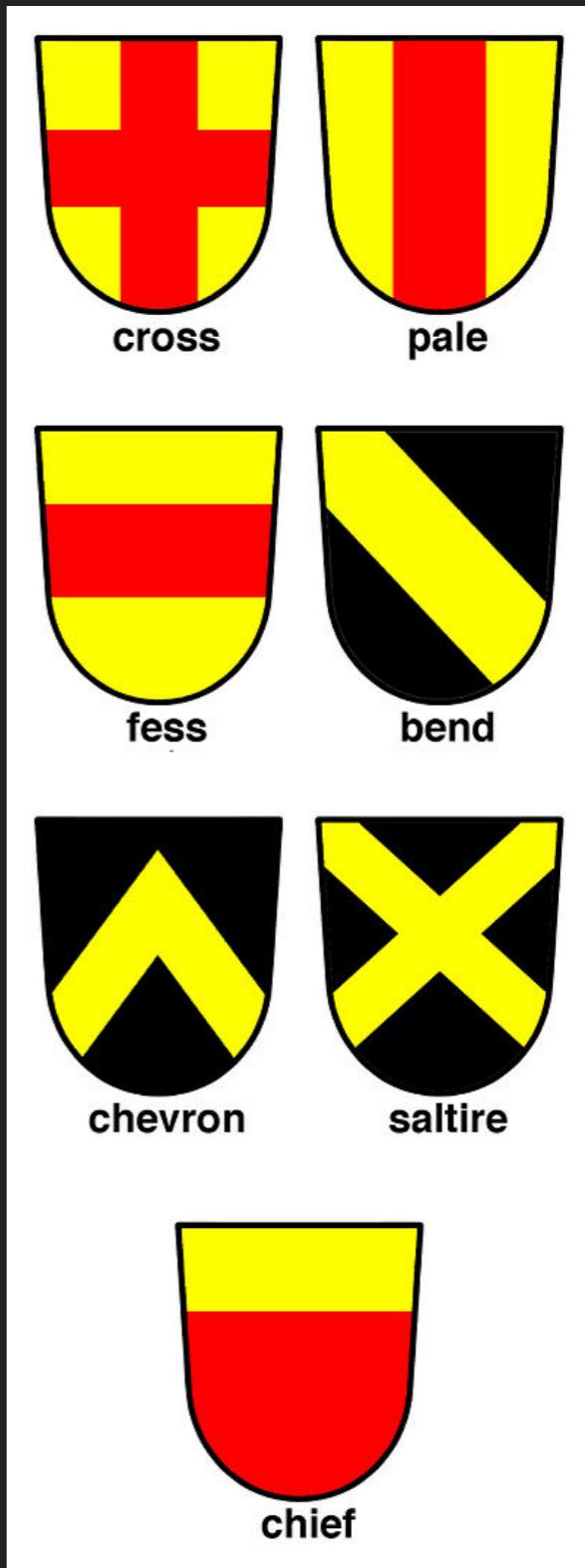
*Vairy or and gules*

```
start
  arms
    field
      tincture
        fur
        |--- fur
        |     vair
        |--- tincture or
        |     and
        |--- tincture gules
```



## ORDINARIES

ORDINARY: "Bend"i // Diagonal Stripe  
| "Chevron"i // Check mark  
| "Chief"i // Stripe on top  
| "Fesse"i | "Fess"i // Horizontal Stripe  
| "Pale"i // Vertical Stripe  
| "Saltire"i // Giant X  
| "Cross"i // Cross



Source: Wikipedia

# SUBORDINARIES

SUBORDINARY: "Pall"i

- | "Pile"i
- | "Pile reversed"i
- | "Quarter"i

// square on the upper left, can have a recursive coat

- | "Canton "i
- | "Bordure"i | "Border"i
- | "Orle"i
- | "Bars"i
- | "Fret"i
- | "Escrutcheon" // Mini shield!
- | "Flaunches"i
- | "Label"i
- | "Gyron"i
- | "Gore"i // Looks like a flipping page
- | "Dance"i // Wavey line

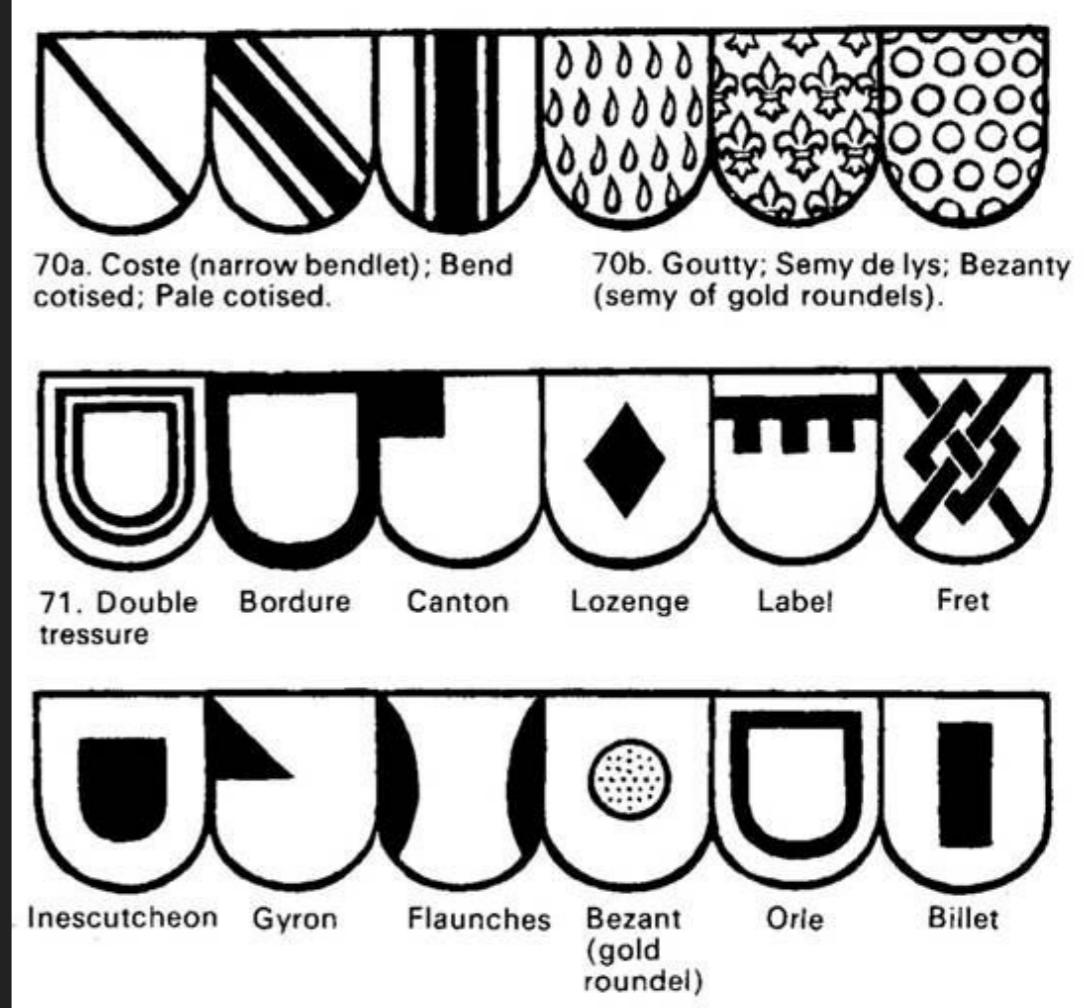
// variations on diamonds

- | "Lozenge"i | "Fusil"i | "Mascle"i

// ball or circle - variants denote color variations

- | "Roundel"i | "Annulet"i
- | "Bezant"i | "Hurt"i
- | "Billet"i // vertical rectangle

... Lots more





Azure, a bend vairy  
or and gules



Sable, a pallet argent,  
2 flanches or

# ARMS OF DALRYMPLE OF WOODHEAD, SCOTLAND



Ermine, on bend  
azure 3 mullets or



Argent; on a saltire gules  
five rustres argent, in chief a  
lion rampant of the second

## TEXT

---

ordinary: ORDINARY | SUBORDINARY

ordinary\_charge: number? tincture? ordinary "s"? tincture? arrangement?

ON: "on"i | "in"i | "all in"i | “within”i

| “charged with” | “charged on”i | “each charged of”i

charge\_group: charge\_group AND charge\_group

| charge\_group “,”? ON subgroup

| ON charge\_group subgroup

subgroup: charge\_group

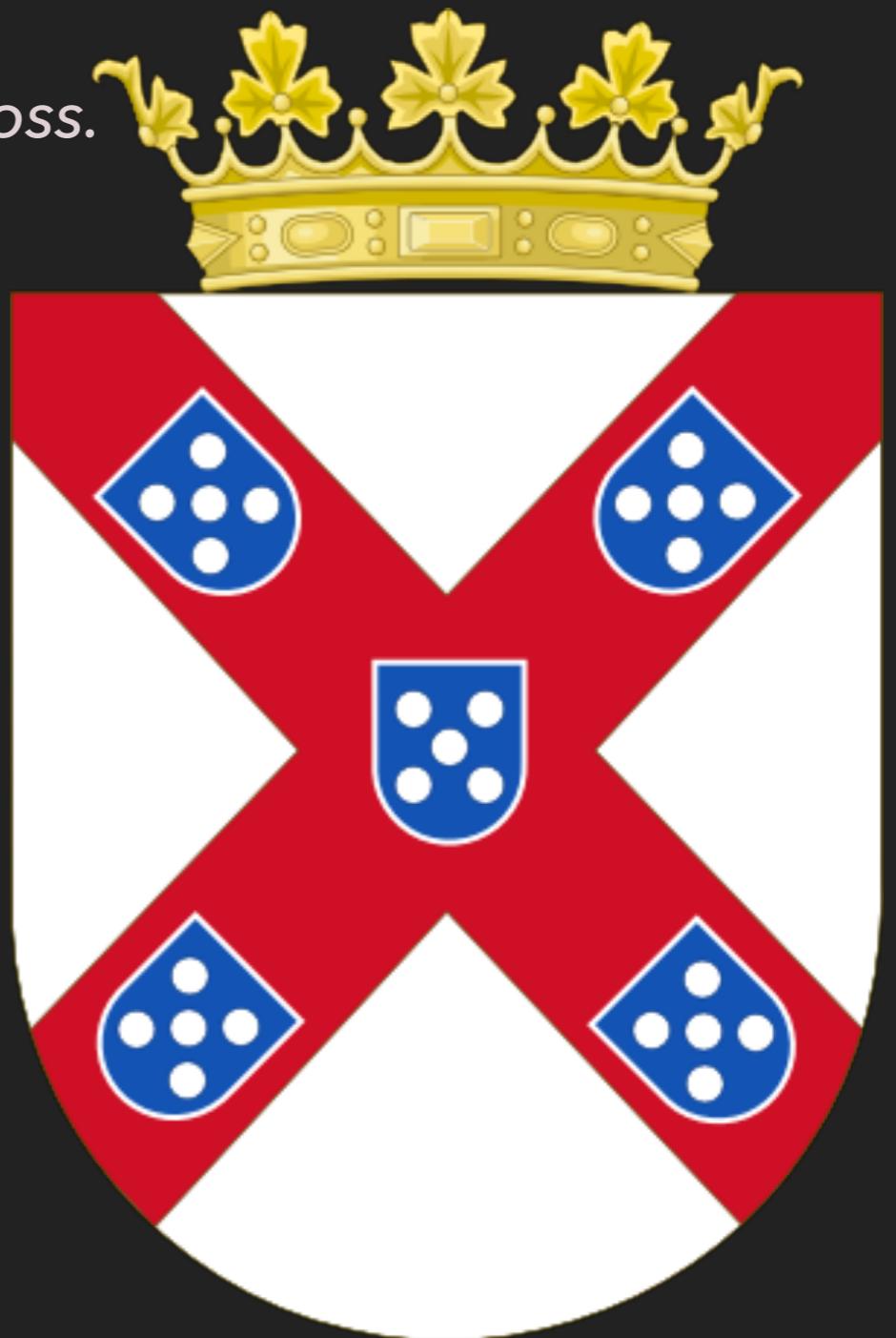
arms: field

| field “,”? charge\_group

| charge\_group

*Argent, a saltire Gules,  
charged of five escutcheons Azure,  
charged each of five bezants Argent, set in cross.*

```
start
  arms
    field
      tincture Argent
    charge_group
      charge_group
        charge_group
          charge
            number a
            ordinary saltire
            attitude
              general_attitude
              tincture Gules
    relation charged of
    subgroup
      charge_group
        charge
          number five
          object escutcheon
          attitude
            general_attitude
            tincture Azure
    relation charged each of
    subgroup
      charge_group
        charge
          number five
          ordinary bezant
          attitude
            general_attitude
            tincture Argent
arrangement
```



**ARMS OF DUKES OF BRAGANZA  
(PORTUGAL)**

## ANIMALS / OBJECTS

animal: HUMAN | BEAST | BIRD | REPTILE

HUMAN: "Man"i | "Men"i | "Old Man"i

BEAST: "Lion"i | "Wolf"i

| "Leopard"i | "Bear"i  
| "Bull"i | "Calf"i  
| "Buck"i | "Stag"i  
| "Dog"i | "Cat"i  
| "Sea-lion"i | "Cockatrice"i

BIRD: "Eagle"i | "Martlet"i

| "Swan"i | "Raven"i

REPTILE: "Serpent"i | "Firedrake"i

| "Salamander"i | "Wyvern"i

OBJECT: "Harp"i | "Fleur-de-lis"i

... And so on, forever for a million objects,  
tools, individual monsters, legendary  
buildings, etc



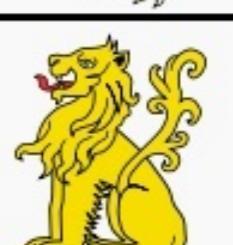
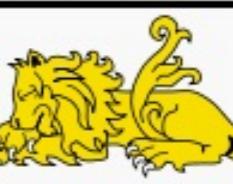
# ATTITUDE

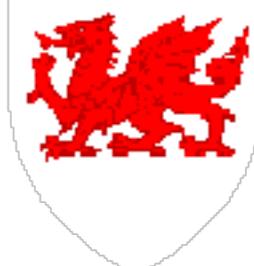
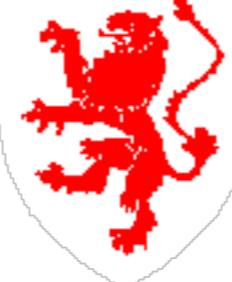
```

attitude: "Rampant"i // Standing on it's hind legs!
| "Passant"i // Walking, one upraised arm
| "Couchant"i // Sitting like a sphinx
| "Gardant"i | "Guardant"i // With head facing the viewer
| "Regardant"i // With head facing behind itself
| "Coward"i // Tail between legs
| "Running"i
| "Sejant"i // Seated
| "Salient"i // Leaping
| "Dormant"i // Sleeping
| "Demi-"i // only the front half of an animal
| "Muzzled"i
| "Combatant"i // Facing each other
| "Addorsed"i // Facing away from each other
| "Tripping"i | "Leaping"i | "Swimming"i
| "Volant"i // soaring
| "Double-headed"i // two headed! cool
| "Erect"i
| attitude AND attitude

charge: number? attitude* _charge_focus "s"? attitude*

```

Attitude	Example
Rampant	
Passant	
Statant	
Salient	
Sejant	
Sejant erect	
Couchant	
Dormant	



haurient

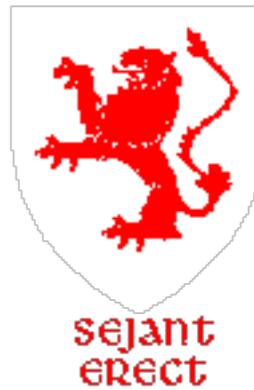
urinant

naiant

embowed

erect

dolphins:

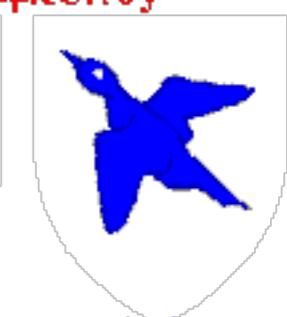
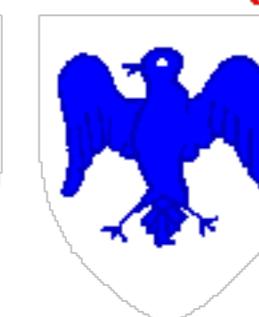
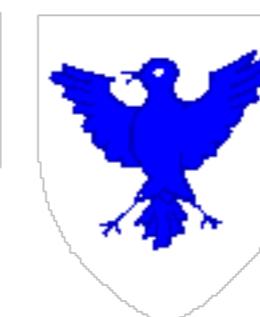
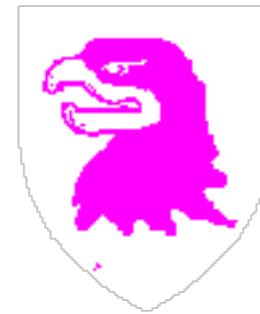
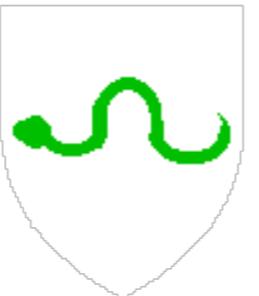
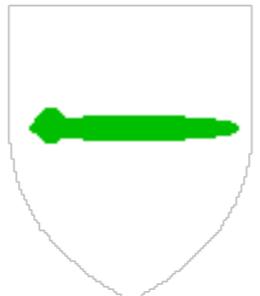
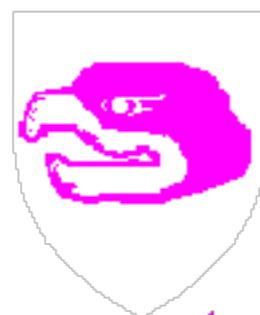
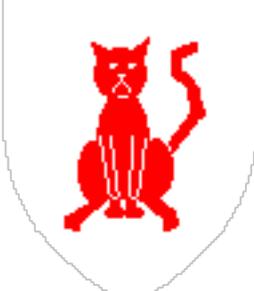
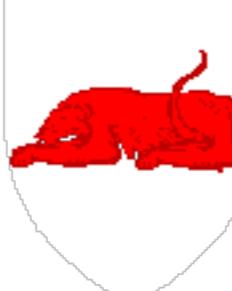
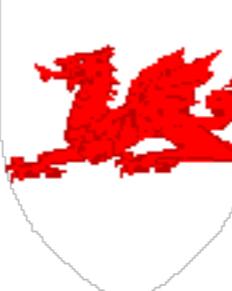


haurient

urinant

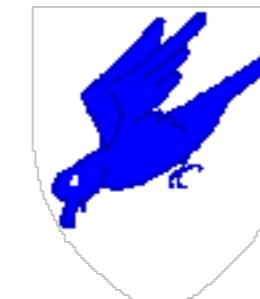
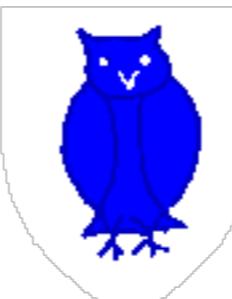
naiant

embowed



couped

erased

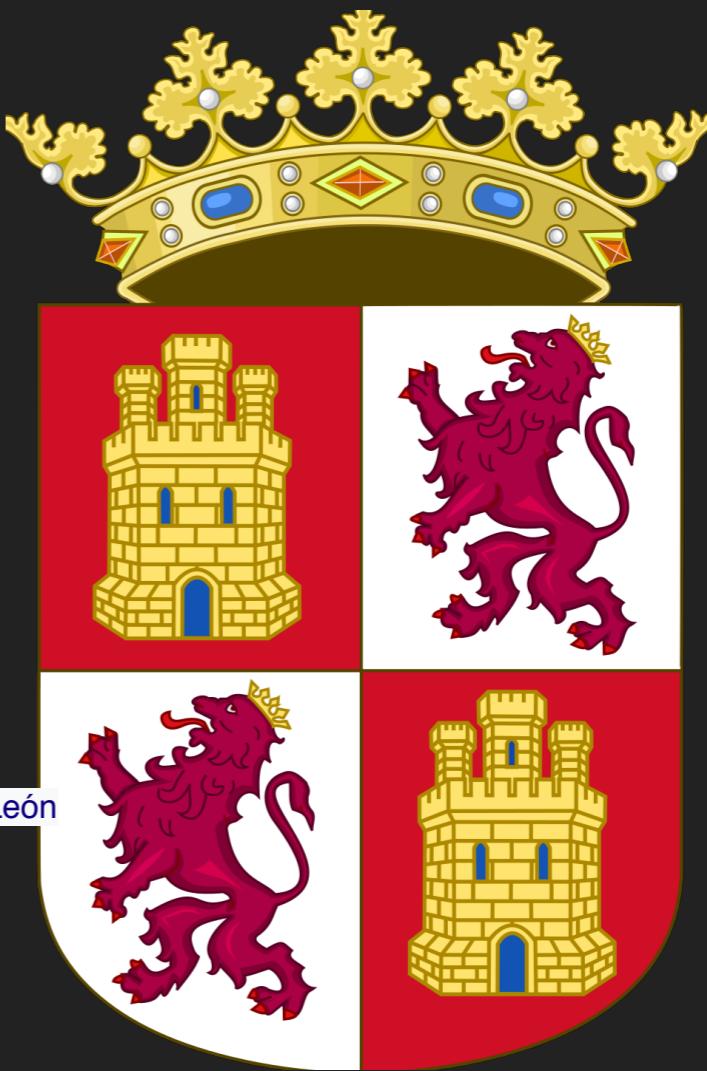


## ARMS OF ROYAL HOUSE OF HOHENZOLLERN



Quarterly **Sable** and **Gules**

## ARMS OF CASTILE AND LEÓN



Quarterly: 1 and 4 Castile, 2 and 3 León

Castile: **Gules** a triple-towered castle **Or**

masoned **Sable** and ajoure **Azure**

León: **Argent** a lion rampant **purpure**, langued and  
armed **gules** crowned **or**

# MARSHALLED ARMS OF EUROPEAN UNION



field: tincture | lines

arms: field  
| field "," charge\_group  
| field charge\_group  
| charge\_group

quarter: (num (AND num)? arms ) COMMENT? ","?

lines: "Quarterly"i ["of"i num] ":"? quarter+  
| "Quarterly"? "Per"i ordinary\_charge quarter+



*Quarterly,*

**first and fourth Gules three Lions passant gardant in pale**

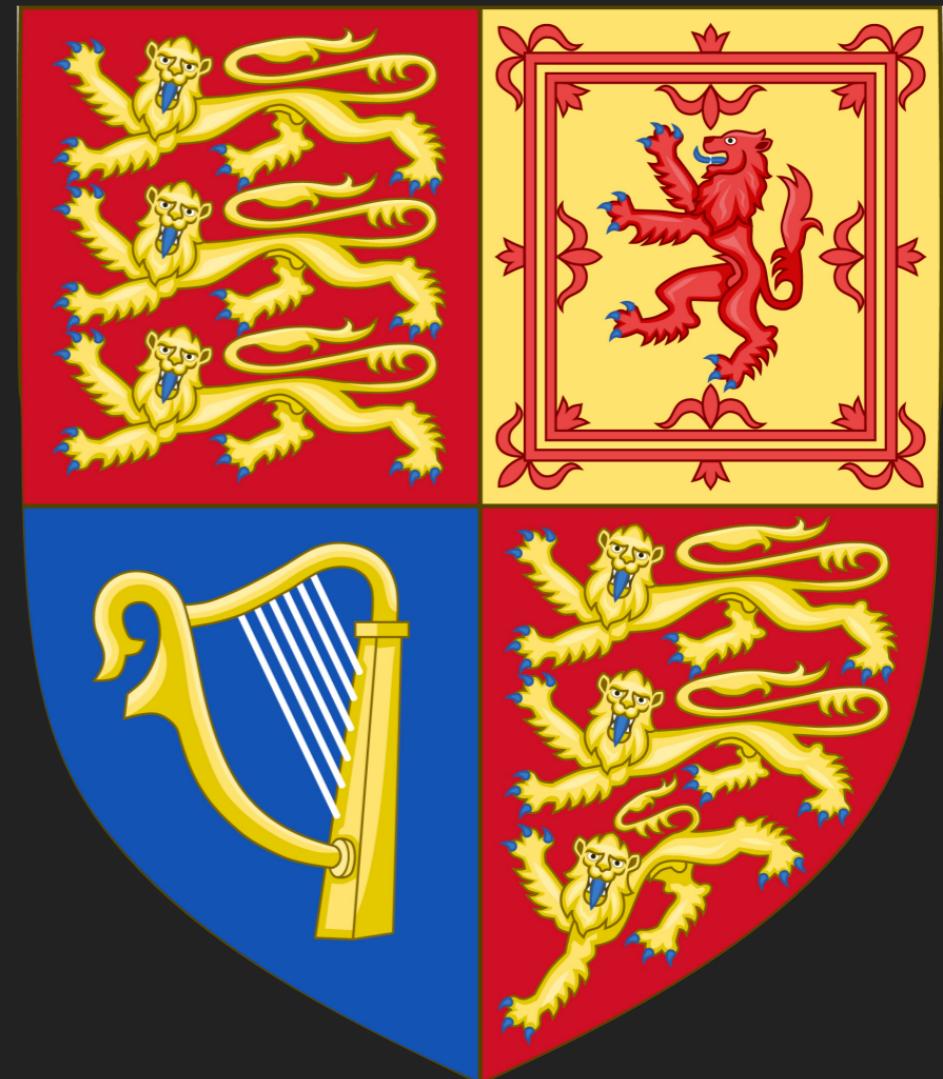
**Or armed and langued Azure (for England),**

*second quarter Or a Lion rampant within a double tressure*

*flory-counter-flory Gules (for Scotland),*

*third quarter Azure a Harp Or stringed Argent (for Ireland)*

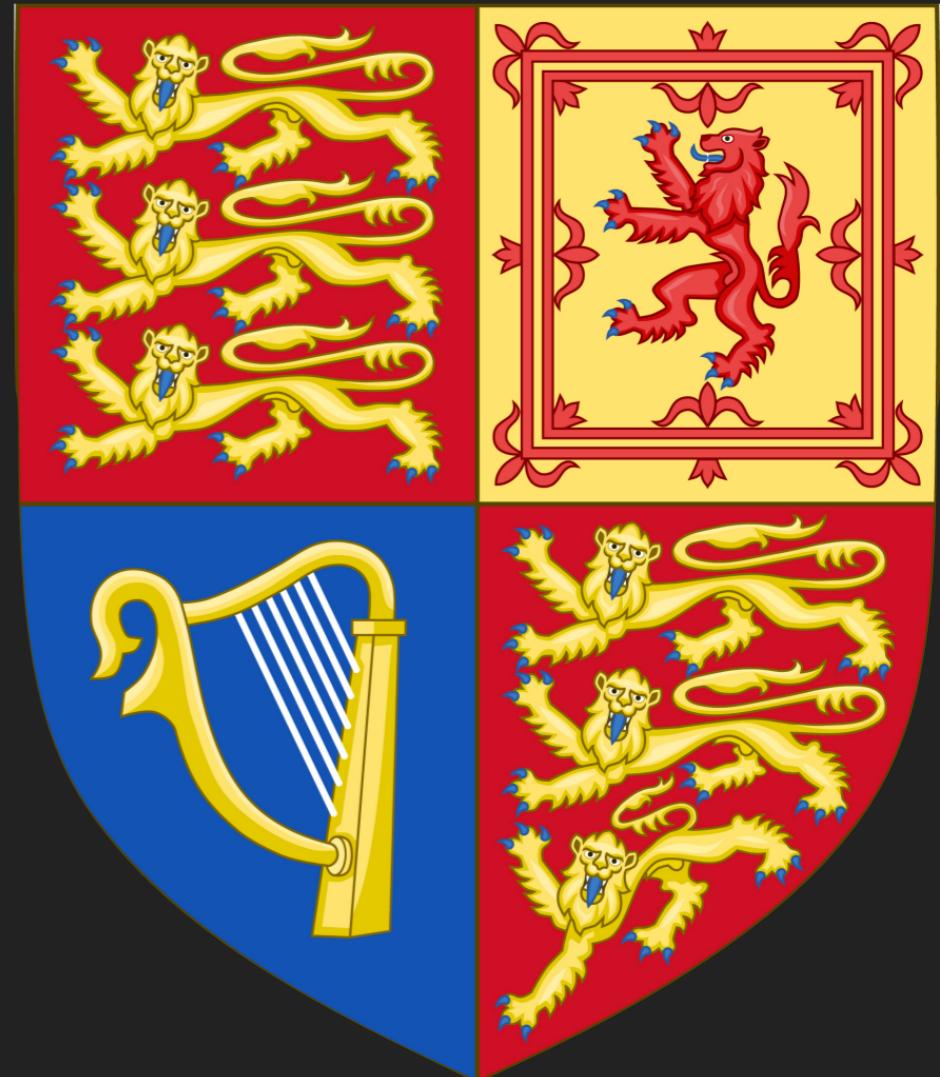
```
start
  arms
    field
      lines
        quarter
          num first
          and
          num fourth
          arms
            field
              lines
                arms
                  field
                    tincture Gules
                    charge
                      num three
                      animal Lion
                      attitude passant
                      attitude gardant
                    ordinary_charge
                    ordinary
                      pale
                      ordinary_attitude
                        general_attitude
                          tincture Or
                        attitude
                          accessory
                            accessory armed
                            and
                            accessory langued
                          tincture Azure
                        (for England)
```



**ARMS OF UK  
(IN ENGLAND)**

*Quarterly,  
first and fourth **Gules** three Lions passant gardant in pale **Or**  
armed and langued **Azure** (for England),  
**second quarter **Or** a Lion rampant within a double  
treasure flory-counter-flory **Gules** (for Scotland),  
third quarter **Azure** a Harp **Or** stringed Argent (for Ireland)***

quarter			
num	second		
arms			
field			
lines			
arms			
field			
tincture	<b>Or</b>		
charge			
num	a		
animal	Lion		
attitude	rampant		
ordinary_charge			
num			
number			
ordinary			
treasure			
ordinary_attitude			
general_attitude			
tincture			
variation			
multipattern		flory-counter-flory	
field			
tincture	<b>Gules</b>		
(for Scotland)			



**ARMS OF UK  
(IN ENGLAND)**

*Quarterly,*

*first and fourth **Gules** three Lions passant gardant in pale **Or** armed and langued **Azure** (for England),*

*second quarter **Or** a Lion rampant within a double tressure flory-counter-flory **Gules** (for Scotland),*

**third quarter *Azure* a Harp *Or* stringed *Argent* (for Ireland)**

quarter

num third

arms

field

tincture *Azure*

charge

num

number a

object Harp

attitude

attitude

attitude

general\_attitude

tincture *Or*

attitude

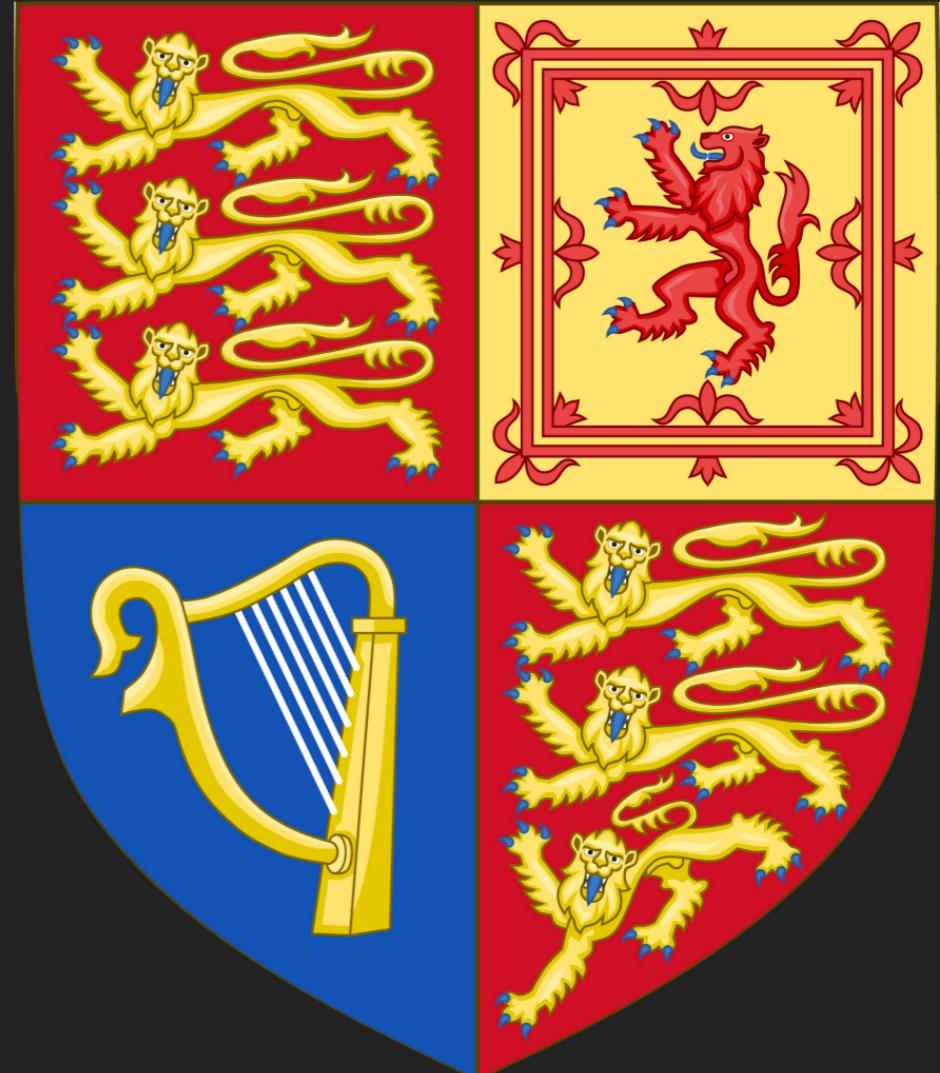
accessory stringed

attitude

general\_attitude

tincture *Argent*

(for Ireland)



**ARMS OF UK  
(IN ENGLAND)**

# RULE OF TINCTURE

## RULE OF TINCTURE



**Sable, bend gules**

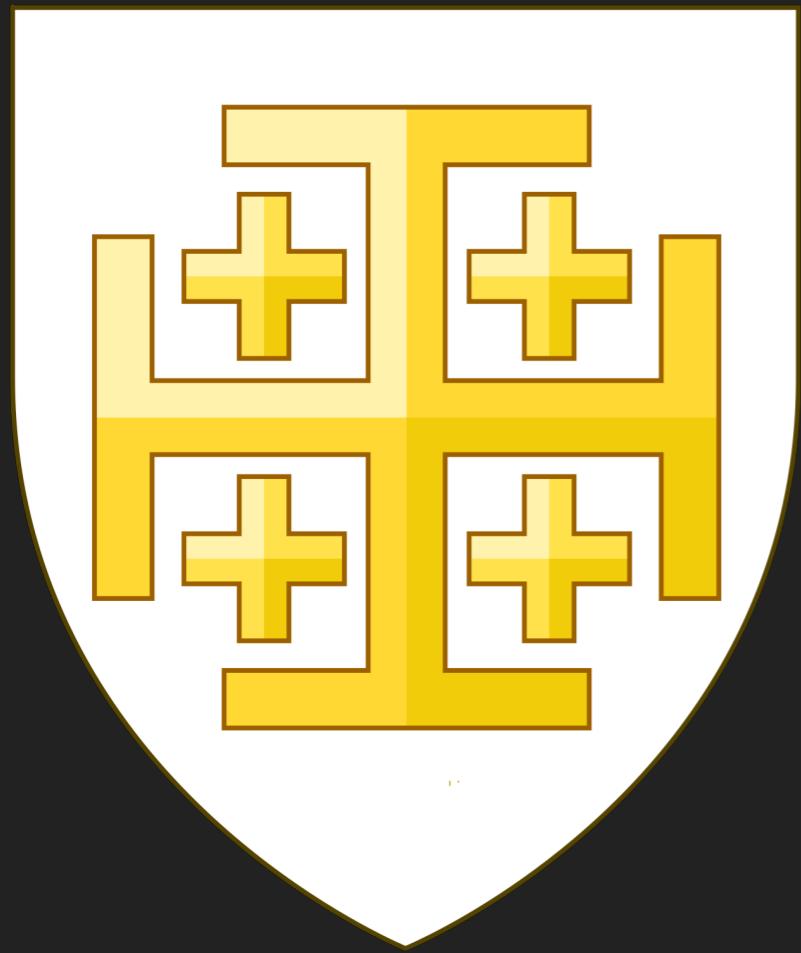
TINCTURE ON TINCTURE  
NOT ALLOWED!



**Sable, bend Or**

METAL ON TINCTURE  
ALLOWED!

## ARMS OF KINGDOM OF JERUSALEM (11TH CENTURY)



Argent, a cross potent  
between four plain  
crosslets Or

METAL ON METAL  
NOT ALLOWED!

## ARMS OF GERMANY (CONTEMPORARY)



Or, an eagle displayed Sable armed  
beaked and langued Gules

TINCTURE ON METAL -  
ALLOWED!

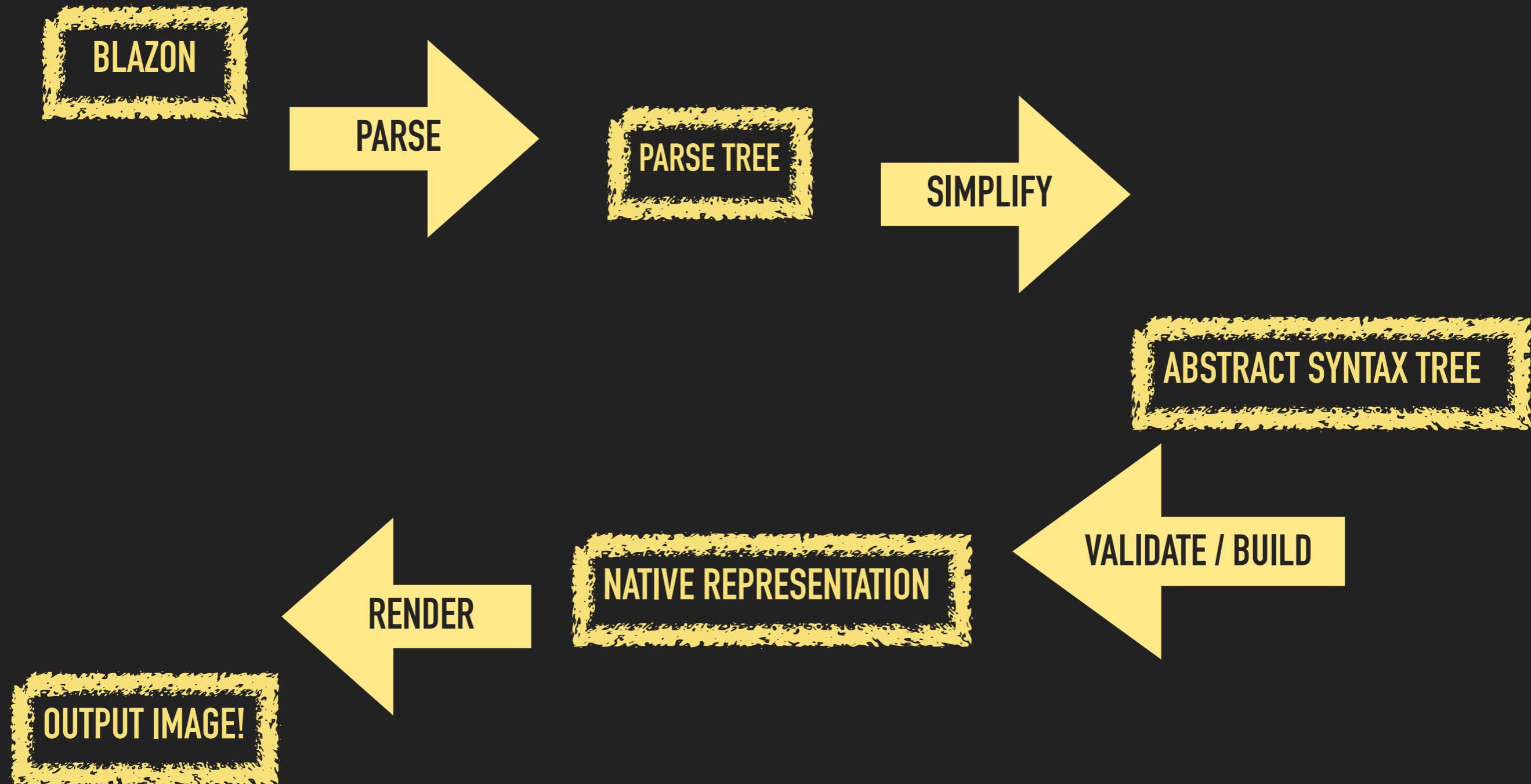
## RULE OF TINCTURE

- ▶ No metal may be on metal, no tincture on tincture
- ▶ Textured/varied fields can have anything on them
- ▶ “Proper” charges can be on anything
- ▶ Applies to some ordinaries but not field divisions
- ▶ More prominent in some countries (Scotland) than others (France)

---

## RULE OF TINCTURE

► Too Complicated for a parser!



# HOMEWORK

# HOMEWORK

GO READ THE  
PYTHON GRAMMAR!



python 3 grammar



All

Images

Videos

News

Shopping

More

Settings

Tools

About 11,500,000 results (0.50 seconds)



## 10. Full Grammar specification – Python 3.7.2 documentation

<https://docs.python.org/3/reference/grammar.html> ▾



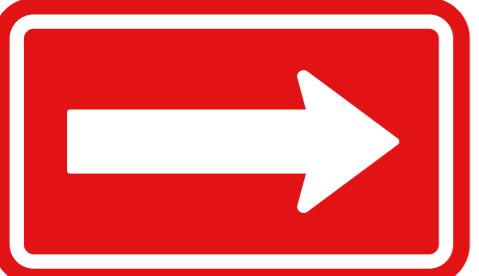
This is the full **Python grammar**, as it is read by the parser generator and used to parse **Python source files**: # Grammar for Python # NOTE WELL: You should ...

# IT'S HALF AS LONG

Google   

All Images Videos News Shopping More Settings Tools

About 11,500,000 results (0.50 seconds)

 [10. Full Grammar specification – Python 3.7.2 documentation](#)   
<https://docs.python.org/3/reference/grammar.html> ▾  
This is the full **Python grammar**, as it is read by the parser generator and used to parse **Python source files**: # Grammar for Python # NOTE WELL: You should ...

# AS MY HERALDRY GRAMMAR!

## PYTHON GRAMMER

```
if_stmt: 'if' test ':' suite ('elif' test ':' suite)* ['else' ':' suite]
```

```
while_stmt: 'while' test ':' suite ['else' ':' suite]
```

## PYTHON GRAMMER

```
if_stmt: 'if' test ':' suite ('elif' test ':' suite)* ['else' ':' suite]  
  
while_stmt: 'while' test ':' suite ['else' ':' suite]  
  
suite: simple_stmt | NEWLINE INDENT stmt+ DEIDENT  
test: or_test ['if' or_test 'else' test] | lambdef  
  
or_test: and_test ('or' and_test)*  
and_test: not_test ('and' not_test)*  
not_test: 'not' not_test | comparison  
comparison: expr (comp_op expr)*
```

## PYTHON GRAMMER

```
stmt: simple_stmt
    | compound_stmt

simple_stmt: small_stmt (';' small_stmt)* '[';']' NEWLINE
small_stmt: (expr_stmt
            | del_stmt
            | pass_stmt
            | flow_stmt
            | import_stmt
            | global_stmt
            | nonlocal_stmt
            | assert_stmt)

compound_stmt: if_stmt
            | while_stmt
            | for_stmt
            | try_stmt
            | with_stmt
            | funcdef
            | classdef
            | decorated
            | async_stmt
```

## PYTHON GRAMMER

```
classdef: 'class' NAME ['(' [arglist] ')'] ':' suite  
  
simple_stmt: small_stmt (';' small_stmt)* ';' NEWLINE  
small_stmt: (expr_stmt | del_stmt | pass_stmt | flow_stmt |  
           import_stmt | global_stmt | nonlocal_stmt | assert_stmt)  
  
compound_stmt: if_stmt | while_stmt | for_stmt | try_stmt | with_stmt |  
funcdef | classdef | decorated | async_stmt
```



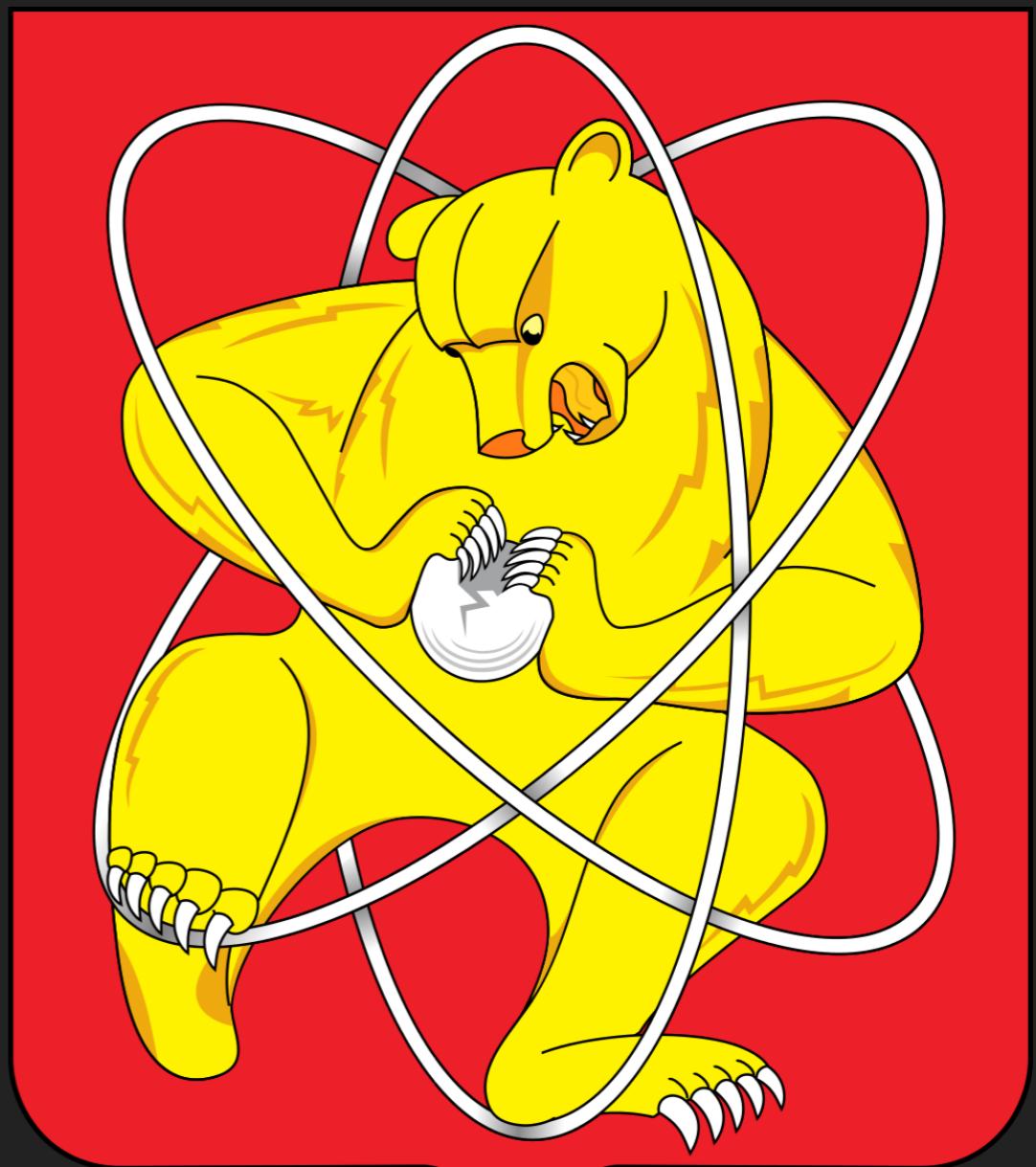
```
print('I <3 Heraldry')
start
eval_input
testlist
funcall
var      print
arguments
string   'I <3 Heraldry'
```

## ANY QUESTIONS

```
import time
if time.time() > 1554510837000:
    print("Talk's over, Any questions?")

<Parsed python code>
file_input
import_stmt
import_name
dotted_as_names
dotted_as_name
dotted_name    time
compound_stmt
if_stmt
comparison
funccall
getattr
var time
time
>
number 1554510837000
suite
funccall
var  print
arguments
string      "Talk's over, Any questions?"
```

## ARMS OF ZHELEZNOGORSK (RUSSIAN PLUTONIUM PRODUCTION TOWN)



HERALDRY RESOURCES:

DRAWSHIELD.NET  
REDDIT /R/HERALDRY  
WIKIPEDIA