
BioArch

Release 0.0.35

May 02, 2020

Contents:

1	bioarch	1
1.1	bioarch package	1
2	Indices and tables	13
	Python Module Index	15
	Index	17

1.1 bioarch package

1.1.1 Submodules

1.1.2 bioarch.age module

```
class bioarch.age.AgeCategory
    Bases: enum.Enum

    An enumeration.

    ADULT = 6
    MIDDLE = 3
    MIDDLE_OLD = 4
    OLD = 5
    UNKNOWN = 0
    YOUNG = 1
    YOUNG_ADULT = 2

    as_quad()

    dtype = <function AgeCategory.dtype>
    parse = <function AgeCategory.parse>

class bioarch.age.EstimatedAge (category: str, ranged: Optional[str])
    Bases: object

    docstring for EstimatedAge

    MAX_AGE = 100
```

```
static empty()  
to_pd_data_frame(index)
```

1.1.3 bioarch.individual module

```
class bioarch.individual.AgeSexStature(osteological_sex: bioarch.individual.OsteologicalSex,  
                                       age: bioarch.age.EstimatedAge, femur: bioarch.individual.LongBoneMeasurement,  
                                       humerus: bioarch.individual.LongBoneMeasurement,  
                                       tibia: bioarch.individual.LongBoneMeasurement,  
                                       stature: Optional[str], body_mass: Optional[str])
```

Bases: object

docstring for AgeSexStature

age

body_mass

static empty()

femur

humerus

osteological_sex

stature

tibia

to_pd_data_frame(index)

```
class bioarch.individual.BurialInfo(site_name: str, site_id: str)
```

Bases: object

docstring for BurialInfo

to_pd_series(prefix="")

```
class bioarch.individual.Individual(_id: str, site: bioarch.individual.BurialInfo,  
                                     age_sex_stature: bioarch.individual.AgeSexStature,  
                                     mouth: bioarch.mouth.Mouth, occupational_markers: bioarch.occupational_markers.OccupationalMarkers,  
                                     joints: bioarch.joints.Joints, trauma: bioarch.trauma.Trauma, context: bioarch.context.Context)
```

Bases: object

docstring for Individual

to_pd_data_frame()

```
class bioarch.individual.LongBoneMeasurement(_max: Optional[float], bi: Optional[float],  
                                              head: Optional[float], distal: Optional[float])
```

Bases: object

docstring for LongBoneMeasurement

static empty()

static empty_lr()

```

    to_pd_series (prefix="")
class bioarch.individual.OsteologicalSex (pelvic: Optional[bioarch.sex.Sex], cranium:
                                         Optional[bioarch.sex.Sex], combined: Op-
                                         tional[bioarch.sex.Sex])
    Bases: object
    docstring for OsteologicalSex
    static empty ()
    to_pd_data_frame (index)

```

1.1.4 bioarch.left_right module

```

class bioarch.left_right.LeftRight (left: Optional[T], right: Optional[T])
    Bases: typing.Generic
    Wrapper around measurements taken on both the left and right sides. This augments the two measurements by
    adding a meta-measurement with the “avg”/“best” combination of both.
    avg () → Optional[T]
        meta-measurement with the “avg”/“best” combination of both.
    left
    right
bioarch.left_right.best_effort_avg (left: Optional[T], right: Optional[T]) → Optional[T]

```

1.1.5 bioarch.mouth module

```

class bioarch.mouth.Mouth (teeth: List[bioarch.mouth.Tooth])
    Bases: object
    Object to hold the 32 teeth
    static empty ()
    to_pd_series (prefix="")
class bioarch.mouth.Tooth
    Bases: object
    docstring for Tooth
    abcess
        'NA', '0', '1'
    calculus
        'NA', '0', '1', '2', '3'
    cavities
        'NA', '0', '1'
    eh
        'NA', '0', '1'
    static empty ()
    to_pd_series (prefix="")

```

tooth

‘NA’, ‘O’, ‘I’, ‘A’, ‘B1’, ‘B2’, ‘C’, ‘D’, ‘E’, ‘F’, ‘G’, ‘H’, ‘T’

1.1.6 bioarch.occupational_markers module

class bioarch.occupational_markers.**EnthesialMarker** (*value: Union[int, float], is_s: bool = False, is_oe: bool = False*)

Bases: object

docstring for EnthesialMarker

as_num() → float

static avg (*left: Optional[EnthesialMarker], right: Optional[EnthesialMarker]*) → Optional[bioarch.occupational_markers.EnthesialMarker]

static parse (*value: Any*) → Optional[bioarch.occupational_markers.EnthesialMarker]


```

class bioarch.occupational_markers.OccupationalMarkers (c_trapezius:
    bioarch.left_right.LeftRight[bioarch.occupational_m
    c_o_deltoid:
    bioarch.left_right.LeftRight[bioarch.occupational_m
    c_o_pectoralis_major:
    bioarch.left_right.LeftRight[bioarch.occupational_m
    c_costoclavicular_lig:
    bioarch.left_right.LeftRight[bioarch.occupational_m
    c_subcalvius:
    bioarch.left_right.LeftRight[bioarch.occupational_m
    c_conoid_lig:
    bioarch.left_right.LeftRight[bioarch.occupational_m
    c_trapezoid_lig:
    bioarch.left_right.LeftRight[bioarch.occupational_m
    s_pectoralis_minor:
    bioarch.left_right.LeftRight[bioarch.occupational_m
    s_levator_scapulae:
    bioarch.left_right.LeftRight[bioarch.occupational_m
    s_triceps_long_head:
    bioarch.left_right.LeftRight[bioarch.occupational_m
    s_trapezius:
    bioarch.left_right.LeftRight[bioarch.occupational_m
    h_subscapularis:
    bioarch.left_right.LeftRight[bioarch.occupational_m
    h_teres_major:
    bioarch.left_right.LeftRight[bioarch.occupational_m
    h_latissimus_dorsi:
    bioarch.left_right.LeftRight[bioarch.occupational_m
    h_pectoralis_major:
    bioarch.left_right.LeftRight[bioarch.occupational_m
    h_deltoid:
    bioarch.left_right.LeftRight[bioarch.occupational_m
    h_coracobrachialis:
    bioarch.left_right.LeftRight[bioarch.occupational_m
    h_supraspinatus:
    bioarch.left_right.LeftRight[bioarch.occupational_m
    h_infraspinatus:
    bioarch.left_right.LeftRight[bioarch.occupational_m
    h_teres_minor:
    bioarch.left_right.LeftRight[bioarch.occupational_m
    h_o_extensor:
    bioarch.left_right.LeftRight[bioarch.occupational_m
    h_o_flexor:
    bioarch.left_right.LeftRight[bioarch.occupational_m
    u_brachialis:
    bioarch.left_right.LeftRight[bioarch.occupational_m
    u_o_pronator_quadrateus:
    bioarch.left_right.LeftRight[bioarch.occupational_m
    u_triceps_brachii:
    bioarch.left_right.LeftRight[bioarch.occupational_m
    u_anconeus:
    bioarch.left_right.LeftRight[bioarch.occupational_m
    u_o_supinator:
    bioarch.left_right.LeftRight[bioarch.occupational_m
    r_biceps_brachii:
    bioarch.left_right.LeftRight[bioarch.occupational_m
    r_supinator:
    bioarch.left_right.LeftRight[bioarch.occupational_m
    r_pronator_teres:
    bioarch.left_right.LeftRight[bioarch.occupational_m

```

Bases: object

docstring for OccupationalMarkers

static empty() → bioarch.occupational_markers.OccupationalMarkers

to_pd_data_frame(index) → pandas.core.series.Series

1.1.7 bioarch.sex module

class bioarch.sex.Sex

Bases: enum.Enum

An enumeration.

FEMALE = 0

FEMALE_ASSUMED = 20

FEMALE_LIKELY = 10

MALE = 100

MALE_ASSUMED = 80

MALE_LIKELY = 90

UNKNOWN = 50

as_bin()

dtype = <function Sex.dtype>

parse = <function Sex.parse>

1.1.8 Module contents

class bioarch.AgeCategory

Bases: enum.Enum

An enumeration.

ADULT = 6

MIDDLE = 3

MIDDLE_OLD = 4

OLD = 5

UNKNOWN = 0

YOUNG = 1

YOUNG_ADULT = 2

as_quad()

dtype = <function AgeCategory.dtype>

parse = <function AgeCategory.parse>

class bioarch.EstimatedAge(*category: str, ranged: Optional[str]*)

Bases: object

docstring for EstimatedAge

```

MAX_AGE = 100

static empty()

to_pd_data_frame(index)

class bioarch.BodyPosition
    Bases: enum.Enum

    An enumeration.

    CROUCHED = 2
    CROUCHED_LEFT_SIDE = 3
    CROUCHED_RIGHT_SIDE = 4
    STOMACH = 5
    SUPINE = 0
    SUPINE_FLEXED_LEGS = 1

    dtype = <function BodyPosition.dtype>
    parse = <function BodyPosition.parse>

class bioarch.CompassBearing
    Bases: enum.Enum

    An enumeration.

    EAST = 2
    NORTH = 0
    NORTH_EAST = 1
    NORTH_WEST = 7
    SOUTH = 4
    SOUTH_EAST = 3
    SOUTH_WEST = 5
    WEST = 6

    dtype = <function CompassBearing.dtype>
    parse = <function CompassBearing.parse>

    to_short_code()

class bioarch.Context (body_position: Optional[bioarch.context.BodyPosition],
                        body_orientation: Optional[bioarch.context.CompassBearing],
                        disturbed: Optional[bioarch.context.Present], decapita-
                        tion: Optional[bioarch.context.Present], double_grave:
                        Optional[bioarch.context.Present], stone_layer: Op-
                        tional[bioarch.context.Present], grave_goods: Dict[str, Optional[Any]])

    Bases: object

    docstring for Context

    static empty()

    static group(value)

    to_pd_data_frame(index)

```

```
class bioarch.Present
    Bases: enum.Enum

    An enumeration.

    NOT_PRESENT = 0

    PRESENT = 1

    dtype = <function Present.dtype>
    parse = <function Present.parse>

class bioarch.AgeSexStature(osteological_sex: bioarch.individual.OsteologicalSex,
                           age: bioarch.age.EstimatedAge, femur:
                           bioarch.individual.LongBoneMeasurement, humerus:
                           bioarch.individual.LongBoneMeasurement, tibia:
                           bioarch.individual.LongBoneMeasurement, stature: Optional[str],
                           body_mass: Optional[str])

    Bases: object
    docstring for AgeSexStature
    age
    body_mass
    static empty()
    femur
    humerus
    osteological_sex
    stature
    tibia
    to_pd_data_frame(index)

class bioarch.BurialInfo(site_name: str, site_id: str)
    Bases: object
    docstring for BurialInfo
    to_pd_series(prefix="")

class bioarch.Individual(_id: str, site: bioarch.individual.BurialInfo, age_sex_stature:
                        bioarch.individual.AgeSexStature, mouth: bioarch.mouth.Mouth, occu-
                        pational_markers: bioarch.occupational_markers.OccupationalMarkers,
                        joints: bioarch.joints.Joints, trauma: bioarch.trauma.Trauma, context:
                        bioarch.context.Context)

    Bases: object
    docstring for Individual
    to_pd_data_frame()

class bioarch.LongBoneMeasurement(_max: Optional[float], bi: Optional[float], head: Op-
                                   tional[float], distal: Optional[float])

    Bases: object
    docstring for LongBoneMeasurement
    static empty()
    static empty_lr()
```

```

    to_pd_series (prefix="")
class bioarch.OsteologicalSex (pelvic: Optional[bioarch.sex.Sex], cranium: Optional[bioarch.sex.Sex], combined: Optional[bioarch.sex.Sex])
    Bases: object
    docstring for OsteologicalSex
    static empty ()
    to_pd_data_frame (index)
class bioarch.JointCondition
    Bases: enum.Enum
    1 - slight/mild 2 - medium 3 - extreme 4 - fused 5 - SCHMORL NODES 6 - FRACTURE
    EXTREME = 3
    FRACTURE = 6
    FUSED = 4
    MEDIUM = 2
    MILD = 1
    NORMAL = 0
    SCHMORL_NODES = 5
    avg = <function JointCondition.avg>
    dtype = <function JointCondition.dtype>
    parse = <function JointCondition.parse>
class bioarch.Joints (shoulder: bioarch.left_right.LeftRight[bioarch.joints.JointCondition][bioarch.joints.JointCondition],
    elbow: bioarch.left_right.LeftRight[bioarch.joints.JointCondition][bioarch.joints.JointCondition],
    wrist: bioarch.left_right.LeftRight[bioarch.joints.JointCondition][bioarch.joints.JointCondition],
    hip: bioarch.left_right.LeftRight[bioarch.joints.JointCondition][bioarch.joints.JointCondition],
    knee: bioarch.left_right.LeftRight[bioarch.joints.JointCondition][bioarch.joints.JointCondition],
    ankle: bioarch.left_right.LeftRight[bioarch.joints.JointCondition][bioarch.joints.JointCondition],
    sacro_iliac: bioarch.joints.JointCondition, c1_3: bioarch.joints.JointCondition, c4_7: bioarch.joints.JointCondition, t1_4: bioarch.joints.JointCondition, t5_8: bioarch.joints.JointCondition, t9_12: bioarch.joints.JointCondition, l1_5: bioarch.joints.JointCondition)
    Bases: object
    docstring for Joints
    static empty ()
    to_pd_data_frame (index)
class bioarch.EnthesialMarker (value: Union[int, float], is_s: bool = False, is_oe: bool = False)
    Bases: object
    docstring for EnthesialMarker
    as_num () → float
    static avg (left: Optional[EnthesialMarker], right: Optional[EnthesialMarker]) → Optional[bioarch.occupational_markers.EnthesialMarker]
    static parse (value: Any) → Optional[bioarch.occupational_markers.EnthesialMarker]

```

```

class bioarch.OccupationalMarkers (c_trapezius: bioarch.left_right.LeftRight[bioarch.occupational_markers.EnthesialM
c_o_deltiod: bioarch.left_right.LeftRight[bioarch.occupational_markers.EnthesialM
c_o_pectoralis_major: bioarch.left_right.LeftRight[bioarch.occupational_markers.E
c_costoclavicular_lig: bioarch.left_right.LeftRight[bioarch.occupational_markers.En
c_subcalvius: bioarch.left_right.LeftRight[bioarch.occupational_markers.EnthesialM
c_conoid_lig: bioarch.left_right.LeftRight[bioarch.occupational_markers.Enthesiall
c_trapezoid_lig: bioarch.left_right.LeftRight[bioarch.occupational_markers.Enthesi
s_pectoralis_minor: bioarch.left_right.LeftRight[bioarch.occupational_markers.Ent
s_levator_scapulae: bioarch.left_right.LeftRight[bioarch.occupational_markers.Ent
s_triceps_long_head: bioarch.left_right.LeftRight[bioarch.occupational_markers.En
s_trapezius: bioarch.left_right.LeftRight[bioarch.occupational_markers.EnthesialM
h_subscapularis: bioarch.left_right.LeftRight[bioarch.occupational_markers.Enthes
h_teres_major: bioarch.left_right.LeftRight[bioarch.occupational_markers.Enthesia
h_latissimus_dorsi: bioarch.left_right.LeftRight[bioarch.occupational_markers.Enth
h_pectoralis_major: bioarch.left_right.LeftRight[bioarch.occupational_markers.Ent
h_deltoid: bioarch.left_right.LeftRight[bioarch.occupational_markers.EnthesialMar
h_coracobrachialis: bioarch.left_right.LeftRight[bioarch.occupational_markers.Ent
h_supraspinatus: bioarch.left_right.LeftRight[bioarch.occupational_markers.Enthes
h_infraspinatus: bioarch.left_right.LeftRight[bioarch.occupational_markers.Enthesi
h_teres_minor: bioarch.left_right.LeftRight[bioarch.occupational_markers.Enthesia
h_o_extensor: bioarch.left_right.LeftRight[bioarch.occupational_markers.Enthesial
h_o_flexor: bioarch.left_right.LeftRight[bioarch.occupational_markers.EnthesialMa
u_brachialis: bioarch.left_right.LeftRight[bioarch.occupational_markers.EnthesialM
u_o_pronator_quadrateus: bioarch.left_right.LeftRight[bioarch.occupational_marks
u_triceps_brachii: bioarch.left_right.LeftRight[bioarch.occupational_markers.Ent
u_anconeus: bioarch.left_right.LeftRight[bioarch.occupational_markers.EnthesialM
u_o_supinator: bioarch.left_right.LeftRight[bioarch.occupational_markers.Enthesia
r_biceps_brachii: bioarch.left_right.LeftRight[bioarch.occupational_markers.Ent
r_supinator: bioarch.left_right.LeftRight[bioarch.occupational_markers.EnthesialM
r_pronator_teres: bioarch.left_right.LeftRight[bioarch.occupational_markers.Ent
r_pronator_quadrateus: bioarch.left_right.LeftRight[bioarch.occupational_markers.E
r_brachioradialis: bioarch.left_right.LeftRight[bioarch.occupational_markers.Ent
f_gluteus_minimus: bioarch.left_right.LeftRight[bioarch.occupational_markers.Ent
f_gluteus_medius: bioarch.left_right.LeftRight[bioarch.occupational_markers.Ent
f_piriformus: bioarch.left_right.LeftRight[bioarch.occupational_markers.EnthesialM
f_obturator_internus: bioarch.left_right.LeftRight[bioarch.occupational_markers.En
f_obturator_externus: bioarch.left_right.LeftRight[bioarch.occupational_markers.En
f_quadrate_femoris: bioarch.left_right.LeftRight[bioarch.occupational_markers.En
f_ilioposas: bioarch.left_right.LeftRight[bioarch.occupational_markers.EnthesialM
f_gluteus_maximus: bioarch.left_right.LeftRight[bioarch.occupational_markers.Ent
f_pectineus: bioarch.left_right.LeftRight[bioarch.occupational_markers.EnthesialM
f_o_vastus_medialis: bioarch.left_right.LeftRight[bioarch.occupational_markers.En
f_o_vastus_lateralis: bioarch.left_right.LeftRight[bioarch.occupational_markers.En
f_adductor_magnus: bioarch.left_right.LeftRight[bioarch.occupational_markers.En
f_o_gastrocnemius: bioarch.left_right.LeftRight[bioarch.occupational_markers.Ent
f_o_plantaris: bioarch.left_right.LeftRight[bioarch.occupational_markers.Enthesial
f_o_popliteus: bioarch.left_right.LeftRight[bioarch.occupational_markers.Enthesial
t_tensor_fascia_latae: bioarch.left_right.LeftRight[bioarch.occupational_markers.E
t_quadiceps: bioarch.left_right.LeftRight[bioarch.occupational_markers.Enthesiall
t_sartorius: bioarch.left_right.LeftRight[bioarch.occupational_markers.EnthesialM
t_gracilis: bioarch.left_right.LeftRight[bioarch.occupational_markers.EnthesialMar
t_semitendinosus: bioarch.left_right.LeftRight[bioarch.occupational_markers.Ent
t_o_tibialis_anterior: bioarch.left_right.LeftRight[bioarch.occupational_markers.E
t_biceps_femoris: bioarch.left_right.LeftRight[bioarch.occupational_markers.Ent
t_semitransversus: bioarch.left_right.LeftRight[bioarch.occupational_markers.En
t_popliteus: bioarch.left_right.LeftRight[bioarch.occupational_markers.EnthesialM
t_o_soleus: bioarch.left_right.LeftRight[bioarch.occupational_markers.EnthesialMa
t_o_tibialis_posterior: bioarch.left_right.LeftRight[bioarch.occupational_markers.E
t_o_flexor_digitorum: bioarch.left_right.LeftRight[bioarch.occupational_markers.E
f_biceps_femoris: bioarch.left_right.LeftRight[bioarch.occupational_markers.Ent

```

```

Bases: object
docstring for OccupationalMarkers

static empty () → bioarch.occupational_markers.OccupationalMarkers
to_pd_data_frame (index) → pandas.core.series.Series

class bioarch.LeftRight (left: Optional[T], right: Optional[T])
    Bases: typing.Generic

    Wrapper around measurements taken on both the left and right sides. This augments the two measurements by
    adding a meta-measurement with the “avg”/”best” combination of both.

    avg () → Optional[T]
        meta-measurement with the “avg”/”best” combination of both.

    left
    right

class bioarch.Mouth (teeth: List[bioarch.mouth.Tooth])
    Bases: object

    Object to hold the 32 teeth

    static empty ()
    to_pd_series (prefix=”)

class bioarch.Tooth
    Bases: object
    docstring for Tooth

    abcess
        ‘NA’, ‘0’, ‘1’

    calculus
        ‘NA’, ‘0’, ‘1’, ‘2’, ‘3’

    cavities
        ‘NA’, ‘0’, ‘1’

    eh
        ‘NA’, ‘0’, ‘1’

    static empty ()
    to_pd_series (prefix=”)

    tooth
        ‘NA’, ‘0’, ‘1’, ‘A’, ‘B1’, ‘B2’, ‘C’, ‘D’, ‘E’, ‘F’, ‘G’, ‘H’, ‘I’

class bioarch.Sex
    Bases: enum.Enum

    An enumeration.

    FEMALE = 0
    FEMALE_ASSUMED = 20
    FEMALE_LIKELY = 10
    MALE = 100
    MALE_ASSUMED = 80

```

```
MALE_LIKELY = 90
UNKNOWN = 50
as_bin()
dtype = <function Sex.dtype>
parse = <function Sex.parse>

class bioarch.Trauma(facial_bones: bioarch.trauma.TraumaCategory, clavicle:
    bioarch.left_right.LeftRight[bioarch.trauma.TraumaCategory][bioarch.trauma.TraumaCategory],
    scapula: bioarch.left_right.LeftRight[bioarch.trauma.TraumaCategory][bioarch.trauma.TraumaCategory],
    humerus: bioarch.left_right.LeftRight[bioarch.trauma.TraumaCategory][bioarch.trauma.TraumaCategory],
    ulna: bioarch.left_right.LeftRight[bioarch.trauma.TraumaCategory][bioarch.trauma.TraumaCategory],
    radius: bioarch.left_right.LeftRight[bioarch.trauma.TraumaCategory][bioarch.trauma.TraumaCategory],
    femur: bioarch.left_right.LeftRight[bioarch.trauma.TraumaCategory][bioarch.trauma.TraumaCategory],
    tibia: bioarch.left_right.LeftRight[bioarch.trauma.TraumaCategory][bioarch.trauma.TraumaCategory],
    fibula: bioarch.left_right.LeftRight[bioarch.trauma.TraumaCategory][bioarch.trauma.TraumaCategory],
    ribs: bioarch.trauma.TraumaCategory, vertebrae:
    bioarch.trauma.TraumaCategory)

Bases: object

docstring for Trauma

static empty()

to_pd_data_frame(index)

class bioarch.TraumaCategory
    Bases: enum.Enum

    An enumeration.

    BLUNT_FORCE_TRAUMA = 6
    BONY_GROWTH = 10
    CRIBA = 5
    FRACTURE = 3
    FUSED = 11
    INFECTION = 2
    NORMAL = 1
    NOT_PRESENT = -1
    OSTEOCHONDritis_DESSICANS = 12
    PARTIAL_BONE = 0.5
    SHARP_FORCE_TRAUMA = 7
    TREPONATION = 8
    UNFUSED = 9
    UNHEALED_FRACTURE = 4

    avg = <function TraumaCategory.avg>
    dtype = <function TraumaCategory.dtype>
    parse = <function TraumaCategory.parse>
```


CHAPTER 2

Indices and tables

- `genindex`
- `modindex`
- `search`

b

- `bioarch`, 6
- `bioarch.age`, 1
- `bioarch.individual`, 2
- `bioarch.left_right`, 3
- `bioarch.mouth`, 3
- `bioarch.occupational_markers`, 4
- `bioarch.sex`, 6

A

abcess (*bioarch.mouth.Tooth attribute*), 3
 abcess (*bioarch.Tooth attribute*), 11
 ADULT (*bioarch.age.AgeCategory attribute*), 1
 ADULT (*bioarch.AgeCategory attribute*), 6
 age (*bioarch.AgeSexStature attribute*), 8
 age (*bioarch.individual.AgeSexStature attribute*), 2
 AgeCategory (*class in bioarch*), 6
 AgeCategory (*class in bioarch.age*), 1
 AgeSexStature (*class in bioarch*), 8
 AgeSexStature (*class in bioarch.individual*), 2
 as_bin() (*bioarch.Sex method*), 12
 as_bin() (*bioarch.sex.Sex method*), 6
 as_num() (*bioarch.EnthesialMarker method*), 9
 as_num() (*bioarch.occupational_markers.EnthesialMarker method*), 4
 as_quad() (*bioarch.age.AgeCategory method*), 1
 as_quad() (*bioarch.AgeCategory method*), 6
 avg (*bioarch.JointCondition attribute*), 9
 avg (*bioarch.TraumaCategory attribute*), 12
 avg() (*bioarch.EnthesialMarker static method*), 9
 avg() (*bioarch.left_right.LeftRight method*), 3
 avg() (*bioarch.LeftRight method*), 11
 avg() (*bioarch.occupational_markers.EnthesialMarker static method*), 4

B

best_effort_avg() (*in module bioarch.left_right*), 3
 bioarch (*module*), 6
 bioarch.age (*module*), 1
 bioarch.individual (*module*), 2
 bioarch.left_right (*module*), 3
 bioarch.mouth (*module*), 3
 bioarch.occupational_markers (*module*), 4
 bioarch.sex (*module*), 6
 BLUNT_FORCE_TRAUMA (*bioarch.TraumaCategory attribute*), 12
 body_mass (*bioarch.AgeSexStature attribute*), 8

body_mass (*bioarch.individual.AgeSexStature attribute*), 2
 BodyPosition (*class in bioarch*), 7
 BONY_GROWTH (*bioarch.TraumaCategory attribute*), 12
 BurialInfo (*class in bioarch*), 8
 BurialInfo (*class in bioarch.individual*), 2

C

calculus (*bioarch.mouth.Tooth attribute*), 3
 calculus (*bioarch.Tooth attribute*), 11
 cavities (*bioarch.mouth.Tooth attribute*), 3
 cavities (*bioarch.Tooth attribute*), 11
 CompassBearing (*class in bioarch*), 7
 Context (*class in bioarch*), 7
 CRIBRA (*bioarch.TraumaCategory attribute*), 12
 CROUCHED (*bioarch.BodyPosition attribute*), 7
 CROUCHED_LEFT_SIDE (*bioarch.BodyPosition attribute*), 7
 CROUCHED_RIGHT_SIDE (*bioarch.BodyPosition attribute*), 7

D

dtype (*bioarch.age.AgeCategory attribute*), 1
 dtype (*bioarch.AgeCategory attribute*), 6
 dtype (*bioarch.BodyPosition attribute*), 7
 dtype (*bioarch.CompassBearing attribute*), 7
 dtype (*bioarch.JointCondition attribute*), 9
 dtype (*bioarch.Present attribute*), 8
 dtype (*bioarch.Sex attribute*), 12
 dtype (*bioarch.sex.Sex attribute*), 6
 dtype (*bioarch.TraumaCategory attribute*), 12

E

EAST (*bioarch.CompassBearing attribute*), 7
 eh (*bioarch.mouth.Tooth attribute*), 3
 eh (*bioarch.Tooth attribute*), 11
 empty() (*bioarch.age.EstimatedAge static method*), 1
 empty() (*bioarch.AgeSexStature static method*), 8
 empty() (*bioarch.Context static method*), 7

empty () (*bioarch.EstimatedAge static method*), 7
empty () (*bioarch.individual.AgeSexStature static method*), 2
empty () (*bioarch.individual.LongBoneMeasurement static method*), 2
empty () (*bioarch.individual.OsteologicalSex static method*), 3
empty () (*bioarch.Joints static method*), 9
empty () (*bioarch.LongBoneMeasurement static method*), 8
empty () (*bioarch.Mouth static method*), 11
empty () (*bioarch.mouth.Mouth static method*), 3
empty () (*bioarch.mouth.Tooth static method*), 3
empty () (*bioarch.occupational_markers.OccupationalMarkers static method*), 6
empty () (*bioarch.OccupationalMarkers static method*), 11
empty () (*bioarch.OsteologicalSex static method*), 9
empty () (*bioarch.Tooth static method*), 11
empty () (*bioarch.Trauma static method*), 12
empty_lr () (*bioarch.individual.LongBoneMeasurement static method*), 2
empty_lr () (*bioarch.LongBoneMeasurement static method*), 8
EnthesialMarker (*class in bioarch*), 9
EnthesialMarker (*class in bioarch.occupational_markers*), 4
EstimatedAge (*class in bioarch*), 6
EstimatedAge (*class in bioarch.age*), 1
EXTREME (*bioarch.JointCondition attribute*), 9

F

FEMALE (*bioarch.Sex attribute*), 11
FEMALE (*bioarch.sex.Sex attribute*), 6
FEMALE_ASSUMED (*bioarch.Sex attribute*), 11
FEMALE_ASSUMED (*bioarch.sex.Sex attribute*), 6
FEMALE_LIKELY (*bioarch.Sex attribute*), 11
FEMALE_LIKELY (*bioarch.sex.Sex attribute*), 6
femur (*bioarch.AgeSexStature attribute*), 8
femur (*bioarch.individual.AgeSexStature attribute*), 2
FRACTURE (*bioarch.JointCondition attribute*), 9
FRACTURE (*bioarch.TraumaCategory attribute*), 12
FUSED (*bioarch.JointCondition attribute*), 9
FUSED (*bioarch.TraumaCategory attribute*), 12

G

group () (*bioarch.Context static method*), 7

H

humerus (*bioarch.AgeSexStature attribute*), 8
humerus (*bioarch.individual.AgeSexStature attribute*), 2

I

Individual (*class in bioarch*), 8
Individual (*class in bioarch.individual*), 2
INFECTION (*bioarch.TraumaCategory attribute*), 12

J

JointCondition (*class in bioarch*), 9
Joints (*class in bioarch*), 9

L

left (*bioarch.left_right.LeftRight attribute*), 3
left (*bioarch.LeftRight attribute*), 11
LeftRight (*class in bioarch*), 11
LeftRight (*class in bioarch.left_right*), 3
LongBoneMeasurement (*class in bioarch*), 8
LongBoneMeasurement (*class in bioarch.individual*), 2

M

MALE (*bioarch.Sex attribute*), 11
MALE (*bioarch.sex.Sex attribute*), 6
MALE_ASSUMED (*bioarch.Sex attribute*), 11
MALE_ASSUMED (*bioarch.sex.Sex attribute*), 6
MALE_LIKELY (*bioarch.Sex attribute*), 11
MALE_LIKELY (*bioarch.sex.Sex attribute*), 6
MAX_AGE (*bioarch.age.EstimatedAge attribute*), 1
MAX_AGE (*bioarch.EstimatedAge attribute*), 6
MEDIUM (*bioarch.JointCondition attribute*), 9
MIDDLE (*bioarch.age.AgeCategory attribute*), 1
MIDDLE (*bioarch.AgeCategory attribute*), 6
MIDDLE_OLD (*bioarch.age.AgeCategory attribute*), 1
MIDDLE_OLD (*bioarch.AgeCategory attribute*), 6
MILD (*bioarch.JointCondition attribute*), 9
Mouth (*class in bioarch*), 11
Mouth (*class in bioarch.mouth*), 3

N

NORMAL (*bioarch.JointCondition attribute*), 9
NORMAL (*bioarch.TraumaCategory attribute*), 12
NORTH (*bioarch.CompassBearing attribute*), 7
NORTH_EAST (*bioarch.CompassBearing attribute*), 7
NORTH_WEST (*bioarch.CompassBearing attribute*), 7
NOT_PRESENT (*bioarch.Present attribute*), 8
NOT_PRESENT (*bioarch.TraumaCategory attribute*), 12

O

OccupationalMarkers (*class in bioarch*), 9
OccupationalMarkers (*class in bioarch.occupational_markers*), 4
OLD (*bioarch.age.AgeCategory attribute*), 1
OLD (*bioarch.AgeCategory attribute*), 6
OSTEOCHONDRITIS_DESSICANS (*bioarch.TraumaCategory attribute*), 12

osteological_sex (*bioarch.AgeSexStature attribute*), 8
osteological_sex (*bioarch.individual.AgeSexStature attribute*), 2
OsteologicalSex (*class in bioarch*), 9
OsteologicalSex (*class in bioarch.individual*), 3

P

parse (*bioarch.age.AgeCategory attribute*), 1
parse (*bioarch.AgeCategory attribute*), 6
parse (*bioarch.BodyPosition attribute*), 7
parse (*bioarch.CompassBearing attribute*), 7
parse (*bioarch.JointCondition attribute*), 9
parse (*bioarch.Present attribute*), 8
parse (*bioarch.Sex attribute*), 12
parse (*bioarch.sex.Sex attribute*), 6
parse (*bioarch.TraumaCategory attribute*), 12
parse() (*bioarch.EnthesialMarker static method*), 9
parse() (*bioarch.occupational_markers.EnthesialMarker static method*), 4
PARTIAL_BONE (*bioarch.TraumaCategory attribute*), 12
PRESENT (*bioarch.Present attribute*), 8
Present (*class in bioarch*), 7

R

right (*bioarch.left_right.LeftRight attribute*), 3
right (*bioarch.LeftRight attribute*), 11

S

SCHMORL_NODES (*bioarch.JointCondition attribute*), 9
Sex (*class in bioarch*), 11
Sex (*class in bioarch.sex*), 6
SHARP_FORCE_TRAUMA (*bioarch.TraumaCategory attribute*), 12
SOUTH (*bioarch.CompassBearing attribute*), 7
SOUTH_EAST (*bioarch.CompassBearing attribute*), 7
SOUTH_WEST (*bioarch.CompassBearing attribute*), 7
stature (*bioarch.AgeSexStature attribute*), 8
stature (*bioarch.individual.AgeSexStature attribute*), 2
STOMACH (*bioarch.BodyPosition attribute*), 7
SUPINE (*bioarch.BodyPosition attribute*), 7
SUPINE_FLEXED_LEGS (*bioarch.BodyPosition attribute*), 7

T

tibia (*bioarch.AgeSexStature attribute*), 8
tibia (*bioarch.individual.AgeSexStature attribute*), 2
to_pd_data_frame() (*bioarch.age.EstimatedAge method*), 2
to_pd_data_frame() (*bioarch.AgeSexStature method*), 8
to_pd_data_frame() (*bioarch.Context method*), 7
to_pd_data_frame() (*bioarch.EstimatedAge method*), 7
to_pd_data_frame() (*bioarch.Individual method*), 8
to_pd_data_frame() (*bioarch.individual.AgeSexStature method*), 2
to_pd_data_frame() (*bioarch.individual.Individual method*), 2
to_pd_data_frame() (*bioarch.individual.OsteologicalSex method*), 3
to_pd_data_frame() (*bioarch.Joints method*), 9
to_pd_data_frame() (*bioarch.occupational_markers.OccupationalMarkers method*), 6
to_pd_data_frame() (*bioarch.OccupationalMarkers method*), 11
to_pd_data_frame() (*bioarch.OsteologicalSex method*), 9
to_pd_data_frame() (*bioarch.Trauma method*), 12
to_pd_series() (*bioarch.BurialInfo method*), 8
to_pd_series() (*bioarch.individual.BurialInfo method*), 2
to_pd_series() (*bioarch.individual.LongBoneMeasurement method*), 2
to_pd_series() (*bioarch.LongBoneMeasurement method*), 9
to_pd_series() (*bioarch.Mouth method*), 11
to_pd_series() (*bioarch.mouth.Mouth method*), 3
to_pd_series() (*bioarch.mouth.Tooth method*), 3
to_pd_series() (*bioarch.Tooth method*), 11
to_short_code() (*bioarch.CompassBearing method*), 7
tooth (*bioarch.mouth.Tooth attribute*), 3
tooth (*bioarch.Tooth attribute*), 11
Tooth (*class in bioarch*), 11
Tooth (*class in bioarch.mouth*), 3
Trauma (*class in bioarch*), 12
TraumaCategory (*class in bioarch*), 12
TREPONATION (*bioarch.TraumaCategory attribute*), 12

U

UNFUSED (*bioarch.TraumaCategory attribute*), 12
UNHEALED_FRACTURE (*bioarch.TraumaCategory attribute*), 12
UNKNOWN (*bioarch.age.AgeCategory attribute*), 1
UNKNOWN (*bioarch.AgeCategory attribute*), 6
UNKNOWN (*bioarch.Sex attribute*), 12
UNKNOWN (*bioarch.sex.Sex attribute*), 6

W

WEST (*bioarch.CompassBearing attribute*), 7

Y

YOUNG (*bioarch.age.AgeCategory attribute*), [1](#)

YOUNG (*bioarch.AgeCategory attribute*), [6](#)

YOUNG_ADULT (*bioarch.age.AgeCategory attribute*), [1](#)

YOUNG_ADULT (*bioarch.AgeCategory attribute*), [6](#)