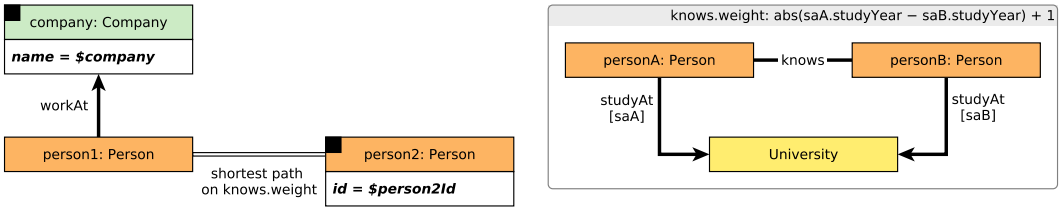
BI / read / 20

BI 1
BI 2
BI 3
BI 4
BI 5
BI 6
BI 7
BI 8
BI 9
BI 10
BI 11
BI 12
BI 13
BI 14
BI 15
BI 16
BI 17
BI 18
BI 19
BI 20

query	BI / read / 20
title	Recruitment
pattern	
desc.	Given a Company `company` and a Person `person2` (who is known to be working at another Company), find a Person (`person1`) working the in the `company` who have the top-20 shortest path to `person2` through people who have studied together. On this path, we only consider edges between Persons who know each other and attended the same university and set the weight of the edge to the absolute difference between the year of enrolment plus 1 (`studyAt.classYear + 1`). If there are multiple Person `person1` nodes with the same shortest path, return all of them.

params

1	company	Long String	Companies with a similar number of employees (former or current) are selected
2	person2Id	ID	

result

1	person1.id	ID	R	
2	totalWeight	64-bit Integer	C	

sort

1	person1.id	↑

limit	20
CPs	3.3, 7.6, 7.7, 8.4, 8.6
relevance	Implementations can either pre-compute edge weights or compute them on-the-fly. To find a weighted shortest path efficiently, implementations can use e.g. a bidirectional Dijkstra algorithm.